Deep Learning-based Food Image Classification and Crowdsourcing-based Calorie Estimation Approach to Support Dietary Management

Patrick McAllister

School of Computing
Faculty of Computing & Engineering
Ulster University

This dissertation is submitted for the degree of

Doctor of Philosophy

I confirm that the word count of this thesis is less than 100,000 words excluding the title page, contents acknowledgements, summary or abstract, abbreviations, footnotes, diagrams, maps, illustrations, tables, appendices, and references or bibliography.

October 2018

Declaration

I confirm that the word count of this thesis is less than 100,000 words excluding the title page, contents acknowledgements, summary or abstract, abbreviations, footnotes, diagrams, maps, illustrations, tables, appendices, and references or bibliography.

Patrick McAllister
October 2018

Acknowledgements

I would like to acknowledge and thank a number of persons who have supported and encouraged me throughout my PhD studies at Ulster University. Firstly, I would like to express my gratitude to my supervisors Professor Huiru Zheng, Dr Raymond Bond, and Dr Anne Moorhead. Without their support, encouragement (and patience!) I would not have completed my PhD studies. I would like to thank Professor Huiru Zheng for her guidance, friendship, and support throughout my PhD studies. I thank Dr Raymond Bond for his help, discussions, and friendship during my PhD studies. I also thank Dr Anne Moorhead for her encouragement in motivating me and providing feedback for each study presented in my Thesis. I would also like to thank PhD students at Ulster University School of Computing for friendship, humour, advice, and for listening to my constant mentioning of Glastonbury Music Festival at any given moment. I also thank Ulster University School of Computing staff for their support during my PhD studies. I especially would also like to express my gratitude to my mother, Mary and father, Michael for their enduring support, patience, and encouragement, while they may not realise this, I owe a lot to them.

Abstract

Food logging is a technique that is used to monitor nutritional intake and research states that food logging is essential for weight management. The aim of this research was two fold; to investigate, develop, and evaluate computer vision and deep learning approaches for food image classification and also to explore and evaluate methods that could be used for predicting nutritional content in food images. In regards to food image logging, this research applied computer vision and deep learning techniques to detect and classify food items in photographs and utilised crowdsourcing along with image processing methods to estimate calories of food portions for dietary management. This research presents studies that inform the development of an automated food image logging platform that combines image classification and calorie estimation to support dietary management. This thesis consists of four studies, Chapter 3 and Chapter 4 research, develop, and evaluate image based approaches and crowdsourcing for calorie estimation. Chapter 5 and Chapter 6 focus on developing and evaluating approaches for food image classification. Chapter 5 investigates the use of conventional image feature extraction approaches with supervised machine learning classification algorithms in classifying a range of food image datasets. Chapter 6 investigates and evaluates the use of deep residual convolutional neural network (CNN) features in classifying a variety of food image datasets. Chapter 7 synthesises results from each study and discusses how they can be integrated into a dietary management framework. This research contributed to the literature by proposing a dietary management system that combines deep learning approaches for food image classification and crowdsourcing for calorie estimation. A novel crowdsourcing calorie adjustment approach was proposed to promote accuracy in food logging for dietary management along with combining state-of-the-art ResNet-152 CNN deep feature extraction with machine learning models to classify variety of food image datasets.

To Oisín, Ché, and Farrah

Table of contents

List of figures

List of tables

Abbreviations

ANN	Artificial Neural Network
API	Application Program Interface
ASM	Angular Second Moment
AUC	Area under the Receiver Operating Characteristic Curve
BMI	Body Mass Index
BoF	Bag of Features
BoVW	Bag of Visual Words
CILMP	Color Intensity Local Mapped Pattern
CNN	Convolutional Neural Networks
CPU	Central Processing Unit
CV	Cross Validation
DFPM	Digital Food Photography Method
DIM	Dimension
DoG	Difference of Gaussian
EFD	Efficient Fractal Dimension
ELM	Extreme Learning Machine
FN	False Negative
FP	False Positive
GFD	Generalised Fractal Dimension
GLCM	Gray Level Co-Occurrence Matrix
GMM	Gaussian Mixture Model
GPU	Graphical Processing Unit
HOG	Histogram of Oriented Gradients
HSV	Hue Saturation Value

IDM	Inverse Different Moment
ILSVRC	ImageNet Large Scale Visual Recognition Challenge
IT	Information Technology
Kcal	Calories
KJ	Kilojoules
LAB	Lightness and *a* and *b* for color components green–red and blue–yellow
LBP	Local Binary Patterns
LoG	Laplacian of Gaussian
MBM	Morphology Based Measures
MKL	Multiple Kernel Learning
MLP	Multilayer Perceptron
NB	Naïve Bayes
NFC	Near-field Communication
PCA	Principal Component Analysis
PFID	Pittsburgh Fast Food Image Dataset
PLF	Pairwise Local Features
PLS	Partial Least Squares
R-CNN	Region Based Convolutional Neural Network
RBF	Radial Basis Function
ReLU	Rectified Linear Units
ResNet	Residual Network
RF	Random Forest
RFID	Radio Frequency Identification
RFPM	Remote Food Photography Method
RGB	Red Green Blue
S-LMP	Sampled Local Map Pattern
SDG	Stochastic Gradient Descent
SFTA	Segmented Fractal Texture Analysis
SIFT	Scale Invariant Feature Transform
SMO	Sequential Minimal Optimisation
SOM	Self Organising Maps
SURF	Speeded-Up-Robust-Features
SVM	Support Vector Machines
TN	True Negative

TP	True Positive
TTBD	Two-Threshold Binary Decomposition
UTM	Unser's Texture Measure
WHO	World Health Organisation
YCbCr	Luminance Chroma Blue Chroma Red

Chapter 1

Introduction

1.1 Background

Obesity is a major health concern globally including in UK and Ireland [1]. It is a term that is used to describe an individual who is excessively overweight [2]. Being obese can have a detrimental effect on an individuals' health as it can cause chronic conditions such as diabetes, bowel cancer, heart disease, and hypertension [2]. The primary cause of obesity is attributed to individuals who consume more calories than they expend via physical activity (PA) [2]. Adults with a Body Mass Index (BMI) greater than 30 indicates that they are obese. The World Health Organisation (WHO) states that 41 million children under the age of 5 years were overweight or obese in 2016 [3]. Furthermore, WHO also states that 340 million children and adolescents that are aged 5-19 were overweight or obese in 2016 [3]. Clinical guidelines state that children, who have a BMI reading that is greater than the 95th percentile for their age group and gender, are considered obese [4]. Statistics state that the rate of obesity has increased from 2.3 to 3.3 fold in the past 25 years in the USA [5]. Different chronic conditions may present themselves to children with obesity such as hypertension, type 2 diabetes, and metabolic syndrome [6,7]. Research has shown that children who are obese tend to become overweight or obese adults [8]. Child obesity within Northern Ireland for 2016/2017, was revealed that 25% of children aged 2-15 were overweight or obese [9].

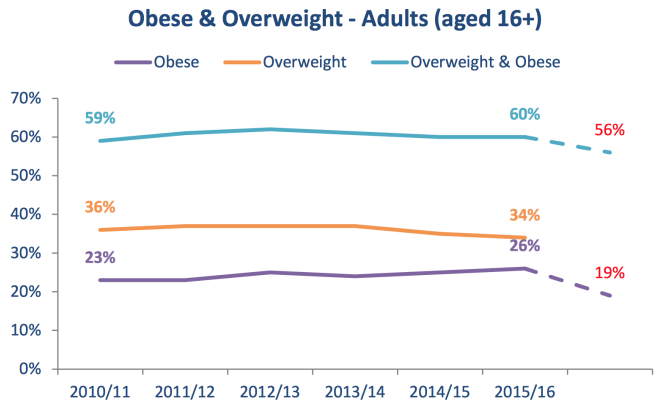


Fig. 1.1 Obesity and overweight trends amongst adults in Northern Ireland [9].

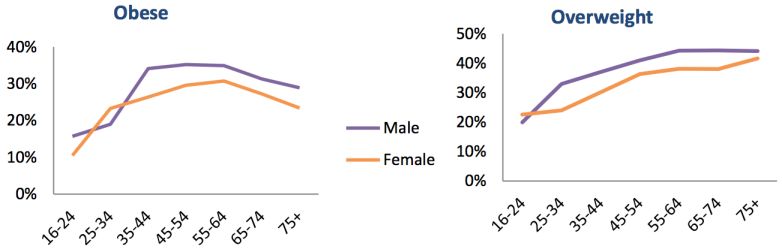


Fig. 1.2 Adult overweight and obesity trends amongst adults in different age groups in Northern Ireland reported in NI Health Survey 2015/2016 report [9].

In England, a Government commissioned report stated reveals that 27% of adults in England are obese and a further 36% are overweight [10]. Further statistics show that in England 68% of adult men and 58% of adult women aged 16 were overweight or obese [10]. In a 2015/2016 NI Health Survey report statistics show that 28% of males and 25% of females were obese. Similarly, the NI Health Survey 2015/2016 Report also states that 37% of males and 32% of females were overweight with the gap between both genders narrowing in the last decade [11]. A more recent survey by Health NI Survey states that over a quarter of adults were classed as obese (27%), and 36% were classed as overweight [9]. Obesity levels show an upward trend when compared to 2006 report, which was 24% [9]. The same survey, also found that a 17% of children aged 2-15 were overweight and 8% were classed as obese. NI Health Survey Report (2013/14) that stated that 61% of adults in Northern Ireland were overweight or obese and the same 2013/14 survey stated that 25% of children are overweight

or obese [12]. In a 2015/2016 NI Health Survey report statistics show that 28% of males and 25% of females were obese [9]. A report in 2012 by Foresight at the Government Office for Science, Tackling Obesities: Future Choices on childhood obesity levels within England suggest that a 14% of people under the age of 20 years will be classed as obese [13]. Current statistics state that nearly one-third of children are overweight or obese and research has highlighted that children are becoming obese younger and staying obese for longer periods of time [14]. These statistics show that obesity is increasing in society and affects all age groups.

1.2 Impact of Diet on Health & Society

Obesity can have a damaging effect on a person's health and well-being and can cause a variety of chronic conditions, e.g. type-2 diabetes, sleep apnea, heart disease, high blood pressure, and some cancers. Obesity can also cause psychological effects on a person's well being such as depression, low quality of life, and anxiety. According to WHO, a raised BMI is a major risk factor for many chronic conditions, e.g. cardiovascular diseases (e.g. stroke and heart disease), cancers (breast, ovarian, liver, and kidney), as well as musculoskeletal disorders [2]. Obesity has increased exponentially at such a scale that it is classed as one of the leading cause of preventable death in the world [14,15] and the rate of obesity is increasing in adults and children. Obesity and being overweight is also linked to more deaths worldwide than underweight [16], and globally more people are overweight than underweight.

Obesity has can also have a significant effect on the wider society and puts a heavy burden on health institutions globally. According to research published in 2016 [6], direct costs to National Health Service (NHS) relating to obesity stand at £6.1 billion annually and costs by 2030 are expected to increase by a further £2 billion. Indirect costs are estimated to

be £27 billion by 2015 [13]. Indirect costs may relate to ill-health or loss of earnings or health conditions caused by obesity [13]. Therefore there is a need to use preventative measures to manage obesity by promoting healthy nutritional food intake and energy expenditure.

	2007	2015	2025	2050
Total NHS costs of diabetes	2.0	2.2	2.6	3.5
Total NHS costs of coronary heart disease	3.9	4.7	5.5	6.1
Total NHS costs of stroke	4.7	5.2	5.6	5.5
Total NHS costs of other related diseases	6.8	7.4	7.8	7.8
Total cost (all related diseases)	17.4	19.5	21.5	22.9
NHS cost increase above current, due to elevated BMI (overweight and obesity)	–	2.1	4.1	5.5
NHS costs attributable to elevated BMI (overweight and obesity)	4.2	6.3	8.3	9.7
NHS costs attributable to obesity alone (see Table 4 in Modelling Future Trends) [1]	2.3	3.9	5.3	7.1
Wider total costs of overweight and obesity, taken at 7x direct costs (figures include rounding effects)	15.8	27	37.2	49.9
Projected percentage of NHS cost @ £70 billion	6.0%	9.1%	11.9%	13.9%

Fig. 1.3 Figure listing the future costs of increased BMI has on NHS in UK (£ billion/year)[6].

The cost of managing increased body mass index (BMI) has risen from each period and is projected to increase into the future [13]. The House of Commons Health Select Committee estimated that the total annual direct cost from obesity and its consequences to the NHS is £5.1 billion per year and for type-2 diabetes costs the NHS £8.8 billion per year [17]. The cost figure was used as a baseline for future estimates in determining the direct cost of obesity on the NHS. As well as direct costs on the NHS, there is also the area of indirect costs on society and government. This may include decreased productivity, which may impose costs on businesses due to absence from work due to high BMI conditions [13]. In regards to the wider research into the area of direct and indirect costs, a review was conducted in [18], and studies were reviewed that focus on the direct and indirect costs of obesity on health services in the period between 2001 and 2011. Results show that the indirect costs are much higher than that of the direct costs between 54% and 59% of total estimated costs [18]. Within these reviews, indirect costs ranged from short-term disability, sickness absence, and lost productivity. Recent research in [19] has stated that obesity represents the second highest impact on the economy with a burden on $73 billion (3% of GDP). Nutritional

intake management is one of the critical approaches (as well as physical activity) to control obesity and promote healthy living. A food diary or food logging can be used to monitor nutritional intake and allows users to be more mindful of what they consume. Research shows that individuals who use a food diary or keep track of nutritional intake lose weight compared to those who do not [20, 21]. Food logging enables individuals to keep track of nutritional eating patterns and highlight behaviours that should be avoided to promote weight loss. Current developments in technology and technology usage rates may be a contributor to an obesogenic environment [22], however, it can also provide an exciting opportunity for weight management and preventative approaches especially in regards to food logging. The following section will discuss various weight management approaches and how technology can be incorporated to promote dietary management especially in regards to food logging.

1.3 Digital Interventions

Technology has become pervasive amongst adolescents as it is estimated that 78.9% of teenagers own a smartphone [23]. Published research also highlight the increase of technology use among all demographics [24].This is an opportunity to use smartphone technologies to create interventions that promote healthy eating and physical activity. Electronic media is shown to improve knowledge of adolescents in nutrition and diet [25]. Behaviour change is a key outcome of using technologies to manage obesity in adults and children, however sustained behaviour change within the long-term scale has proven to be difficult [26, 27]. Obesity can have serious implications on the individual's health and with regard to technology usage statistics, it shows that there are opportunities in which these technologies can be harnessed in order to promote self-management of obesity.

mHealth

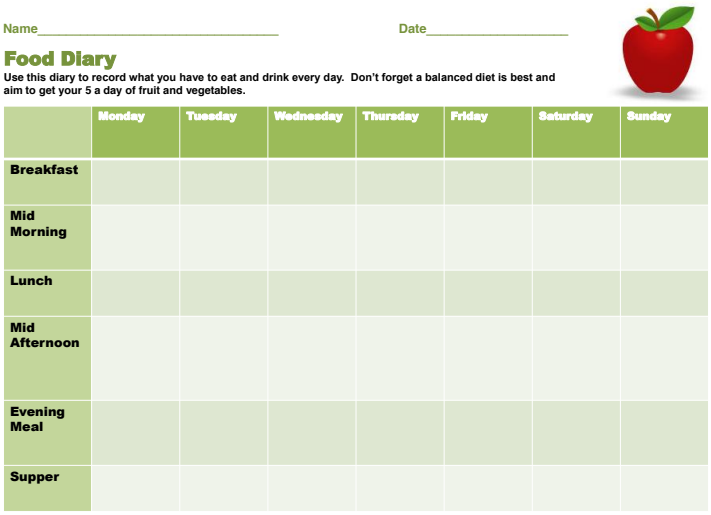
Mobile technology has become ubiquitous and is an essential tool for many in managing their health and diet. The progressive nature of mobile technology has allowed it to be used to manage a users health through data capturing and recording methods. Statistics show that 70% of UK adults use a smartphone device and 93% of 16-24 years old own a smartphone device [24]. Statistics show that 51% of children aged 5-15 have access to a tablet at home, and 62% of 12-15 year old own a Smartphone device [24]. This technology adoption rate allows us to target these individuals by developing mHealth applications that can utilise the devices built into mobile devices to promote dietary management. A vast amount of research has been completed in researching the use of mobile applications to address lifestyle, diet, and activity factors to promote a healthy lifestyle. For example, in [21] a smartphone application was developed that incorporates behavioural change strategies such as goal setting, self-monitoring, and a feedback stage to promote weight management. In the application, daily energy targets are generated using the user's personal information such as height, weight. The user can log their daily food intake. This intake is deducted from their total energy allowance. Goal setting is introduced in the application by generating a daily target goal. This application also generates a weight loss goal. Focus groups were organised to inform the development of the application. Results from these focus groups suggested that users found the application as an encouraging way to manage weight. Other users saw the application as 'disciplinarian' to help them keep on a weight management track. Another mobile application is proposed that monitors the eating habits and physical activity of the user [22]. This application can use this information to offer advice to maintain a healthy lifestyle. The proposed application can determine behaviours among users by analysing the collected data. Bioelectric impedance analysis is used to analyse body fat measurement of the user. The proposed application collects data relating to exercise by using accelerometers and pulse monitors. Once data is collected, it is then analysed to provide advice relating to user's health and to provide a

course of actions. Recent research completed by Hutchesson, et al. also explored the use of using smartphone technologies for self-monitoring dietary intake compared with paper-based methods, and website based approaches. Results show computer and smartphone-based methods were more acceptable approaches for self-managing dietary intake compared to traditional paper-based methods [28].

1.4 Food Logging for Dietary Management

Dietary self-monitoring is a method in which users can document activities and monitor their behaviour for analysis and feedback to promote behaviour change. In regards to nutritional intake, food logging is a self-monitoring approach that consists of documenting dietary intake, and it is widely accepted that food logging or food journaling is an important activity to undertake when trying to promote weight loss or maintain current weight [20]. Much research has been conducted in developing food logging techniques to ensure the experience is convenient for the user and to promote usability [29]. Research has explored the use of food logging, research by Baker et al. suggest that self-monitoring is necessary for weight control after analysing results of participants who monitored their dietary intake for 18 weeks. Results show monitoring dietary intake was positively correlated with weight change and no monitoring at all was negatively correlated with weight change [30]. Similar research by Peterson, et al. examined the effect of constant self-monitoring frequency has on weight change [31]. The main contribution of [31] was examining how the frequency and comprehensiveness correlated with weight loss and results show that increased monitoring frequency yielded beneficial effects in regards to weight change [2].

It has been reported that decreased adherence and retention rate was a problem in regards to maintaining consistent food logging. This problem was experienced in [32], in which 212 participants were asked to use a smartphone weight loss application. Results



Name_____ Date_____

Food Diary

Use this diary to record what you have to eat and drink every day. Don't forget a balanced diet is best and aim to get your 5 a day of fruit and vegetables.

	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Breakfast							
Mid Morning							
Lunch							
Mid Afternoon							
Evening Meal							
Supper							

Fig. 1.4 Example food diary for recording nutritional intake for dietary management.

of [32] experience a decrease in logins after the first month. Low adherence is a familiar trend in some research in regards to dietary self-monitoring. Research completed by Duncan et al. explored the use of an IT-based intervention to improve the dietary behaviours, physical activity, health literacy of middle-aged male participants [33]. In regards to dietary behaviours, participants were asked questions which relates to different types of foods that were consumed during the week and a score was calculated based on the participant responses. Low retention rates were also observed in [33], and similar works also experienced low retention and adherence rates in [34], and low attrition rates is a known problem with ICT based interventions [35]. However, when comparing food logging adherence using information technologies to conventional methods or paper-based formats, improved retention and adherence rates are reported when participants use smartphone-based/ web-based technologies [36, 37].

In [38] research was carried out that researched the main barriers to using a weight loss, dietary self-management application. In [8] weight loss forums, MyFitnessPal Facebook group, and mailing lists were used to research the disadvantages and barriers associated with using a food logging diary application. From this research a number of barriers were

noted, (1) individuals felt that counting calories and recording them into a food logging system is a laborious activity and requires 'too much effort'. Reliable food entry was a common theme expressed by respondents and this related to areas such as the difficulty of entering in energy intake that is reliable. Modern food logging applications consist of a database application programming interface (API) that allows users to search for calorie content. However, this may be incorrect and is not an accurate representation of the food the user has eaten [38]. Therefore, there is a question of database reliability in regards to utilising this functionality. In regards to removing complexity, research has been completed in using a simplified food logging framework [29], the user would log their dietary intake by using high-level food groups, e.g. grains, fruit juice, whole grains, etc. Other reasons for low adherence to food logging is the variety of food in which people eat. Food logging can become very complicated if a user to recalling from memory about the food items eaten over a day [38]. This would lead to inaccurate food logs and affect overall dietary management through overestimations or underestimations. There is also evidence to suggest that greater the size of a meal, the likely the user is to underestimate and that piecewise calorie estimation of individual food items within a food portion can lead to more accurate calorie estimations [39]. Research also states that the average non-expert (non-expert meaning users who are not dieticians or nutritionists) self-reporting can have an error rate that of 400 calories per day [40].

Accurately identifying calories within a food portion is also a major challenge and in [38] it was reported from users that lack of personalisation in regards to calorie estimation is a barrier and the lack of this feature may prevent attrition of an application. As stated, most commercial applications utilise an online API to allow users to search for calorie estimations of food items. Other research utilise barcode readers and near-field communication (NFC) readers to scan the calorie content of food packaging to estimate portion size [41]. A more novel approach would be to utilise crowdsourcing "wisdom of the crowds" as a means

to determine calorie content. This approach has been investigated in [40]. Authors of [40] propose PlateMate which uses crowdsourcing techniques to determine calorie content and portion size using Amazon Mechanical Turk. This platform allows different users to complete tasks of analysing an image of a meal and to use 'wisdom of the crowds' to attain feedback on the nutritional composition. Results from this research show that PlateMate, utilising crowdsourcing, is nearly as accurate as a trained dietician. Further research using crowdsourcing to determine calorie content needs to be investigated in how it can provide support in dietary management. In order to mitigate incorrect food logging, images have been used as a means of reminding the user of nutritional content in a meal. Commercial applications and research have used food image logging as a means for dietary management, but there is still the issue of logging the food items individually to ascertain calorie content which can lead to inaccuracies even using the food image as a reminder. Other research has utilised remote food photograph method (RFPM) to allow nutritionists to remotely analyse the image and provide feedback (Figure 1.6) [42], however, there is still a need for automated analysis of an image to provide nutritional feedback for users regarding correctly identifying food items and providing nutritional analysis of images. It is clear that food logging can promote dietary management through enabling the individual to become mindful of calorie consumption. The pervasive nature of smartphone technologies allows users to use web technologies and food logging applications to quickly document nutritional intake using online calorie databases or more novel methods of determining calorie content using computer vision methods, which will be discussed more in Chapter 2.

In other research, a mobile phone application was designed that concentrated on reducing the amount of time needed for users to create daily food logs [29]. This mobile application records nutritional intake differently from [29] in the mobile application intends to streamline the logging process by breaking down food items into different categories (e.g. dairy, protein, vegetables etc.). This method is very different in comparison to merely

searching for foods using a local or online data. The application employs a scoring algorithm derived from the Healthy Eating Index which is based on USDA 2005 Diet Guidelines. A score of 100 is generated that closely matches what the guidelines state. The user can specify food portion type by selecting '+1' as input, and a score would then be generated. Evaluation results suggested that users preferred using the traditional food logging method as well as the streamlined approach to analysing nutritional intake. To inform the development of this application, a situation-based design approach was adopted. Questionnaires and interviews were organised to find out the requirements for the application. This application takes a different approach to other previously discussed mHealth interventions as recipes are recommended to the user using ingredients found at home. The recipes are tailored as the application takes into account the eating habits. Once the user selects different ingredients and recipes, the application will give the user feedback on these ingredients concerning how healthy they are. The application also allows users to create a photo diary by uploading images of meals and the ingredients. Using food images for dietary management allows for increased analysis in determining accurate nutritional intake. Using images also have the potential to improve dietary recall in determining portion sizes, for example, research completed by Naska, et al. using images of weighted food portions and asked participants to match the food image to the correct portion size [44]. Results show that participants selected the correct or adjacent image to match with the portion size in 90% of the instances. Further results also show that participants tended to overestimate the small food portions and underestimation large food portions. Research presented in [44] show that digital food image photography is promising in helping users determine food portion size for accurate dietary management. Other research suggests that using images may lead less misestimation in food portion sizes for dietary recall [44]. Similar research completed by Valanou, et al. explored the use of food photography in assessing dietary intake of parents of children [45]. The study focused on determining if parents were able to correctly estimate the size of food

portions eaten by children and results show that participants were able to indicate correct food portion size in 97% of the assessments [45]. This research discussed in this section show that smartphone technologies can play an important role in dietary management and the literature shows that various approaches have been utilised in using mHealth technologies. The use of images has also been explored for automated food logging. The following section will briefly discuss different approaches that specifically utilise food image photography for dietary management.

1.5 Food Images for Dietary Management

Recent years have seen the increase in using images to document food intake for dietary management. The use of images are used to address the problems of traditional methods of food logging, and the shortcomings of conventional food logging methods have been described in [38] as being tedious or complicated, these methods include recording food intake using text or using a search option within a smartphone application to search a database or food table for nutritional content. However, the use of images offer a more convenient approach to food logging and provide objective evidence to support dietary management. Using images to provide for dietary management allow the user to avoid problems that are commonplace in text-based food logging methods, problems such as underestimation and also forgetting to document energy intake. Individuals can further analyse portion sizes and to derive more information about food. Some image food logging interventions also ask the user to take an image of the food meal before and after to also increase objectivity and accuracy as to what has been consumed. Some dietary management interventions also use video capture to document food intake. This method also gives the advantage of accurately determining portion size through food dimension and depth. The pervasiveness of smartphone usage enables users to make use of high-pixel cameras to document food intake and allows for

detailed contextual analysis of food intake. The use of food logging applications has increased over the past 10 years due to the popularity of smartphone devices. These applications allow the user to manually enter their food items to document nutritional intake by either using a search menu that is able to search a food API to determine nutritional content, typing in the food item along with calories, or more novel methods such as using a barcode scanner to scan the food item packaging to determine nutritional content. In regards to using images to document food intake, a review [46] of dietary management applications were analysed from Google Play Store, and iOS App Store and 13 applications were identified. Of the 13 applications analysed, only 2 allowed the user to save a photo. In regards to the review in [46] identifying food portion size, image-based examples did not take into account food portion sizes, but only text-based guidelines were available across the applications [46]. Research has shown that pictures of example portion sizes, along with the ability to save photographs of a users meal image, would increase the potential of accurately determining nutritional intake as shown in [47]. In [48] research was also carried out that analysed the adherence of a photograph food logging application using 189,770 participants, users were able to rate the healthiness of meal images of their peers. Results show that there was a low adherence with 2.58% of the participants used the application actively.

Research has been completed in highlighting the advantages of using images for food logging, in [38] the barriers for food logging are discussed. Qualitative analysis was completed that highlighted key challenges from analysing community forum posts for mobile food journals. One hundred and 40 food journalers (mixture of lapsed and active journalers). Common themes from the forum analysis responses in [38] was that the process of food logging was described ‘tedious’, and that it was ‘no fun logging’. Further analysis of forum responses stated that the process of food logging was ‘time-consuming’ and journalers highlighted difficulties relating to personalisation of nutritional intake as calorie content was not available for the foods that the ate. Other food journal users also stated that it was

difficult to determine correct portion sizes [38]. Research in [38] suggested the use of using photo-based food diary as a way to mitigate some of the challenges and limitations of manual food logging. Other research also surveyed 257 people in regards to what they feel are the barriers to food logging and their experiences. Results from [49] suggest that difficulty of food logging is a barrier to continued food logging and 26% of respondents stating that food logging was time-consuming. Other results of the survey in [49] stated that 98% percent of individuals that had food journaling experience using technology reported longer journaling periods (7—12 months median) compared to food journalers that used paper (median 2-3 months). Research in [36] also suggests similar findings that technology-based food logging methods retain users more than paper-based diary methods.

The use of food images have also extended to social media websites such as Instagram, Facebook, and FoodSpotting enable individuals to share their meals with people and has the potential in allowing individuals to become more mindful of meal quality and portion size. In [51] research was completed that explored the use of social media image sharing website Instagram in promoting dietary management through tracking food intake and peer support. Interviews were conducted with 16 people who share their meal images on Instagram to discuss the challenges and benefits of using a social media platform to help reach goals. Users who were interviewed used Instagram to document their nutritional intake by uploading their food and some users uploaded food items to remind themselves of what they ate. The study also highlighted the advantages of using images for dietary management by allowing for more detailed analysis concerning portion size. Another advantage using social media to upload food images is to obtain a visual summary of nutritional intake, e.g. Figure 1.5 depicts the tiles interface used in Instagram. The study also highlighted the advantages of crowdsourcing peer support in posting food meal images to help encourage others to post their food images regardless of nutritional quality [51]. Using images has been shown to enhance dietary management through allowing users to increase the objectivity and accurately determine food

intake. Other research has utilised crowdsourcing approach with food images to determine nutritional content or healthiness of meals [40, 191]. Results indicate that crowdsourcing has potential to inform users of nutritional quality of meals in food images, however more investigation is needed in utilising crowdsourcing to accurately estimate calories in food photography.

Users can analyse their nutritional intake more accurately by determining portion sizes as well as using the meal images as a memory tool. Users can find nutritional content of individual meal items in food images by using the meal image to remind the user and research has suggested that determining the calories in individual meal items instead of a meals entirety can enhance the accuracy [51]. Other research also explores the use of social media influence and persuasive user interaction to rate the healthiness of food images uploaded by users [52]. The work in [53] utilised social cognitive theory to influence users to select healthy choices. Results of the work presented in [53] suggested that individuals were 'slightly' more accountable due to other users being able to view their food choices. Other research was completed in using social media networks as food diaries [54]. Participants were able to use semi-public private groups to document their food intake through listing the ingredients as well as supplying an image of the meal. Images were used to elicit feedback from other users for motivation and to allow the user to reflect on eating habits.

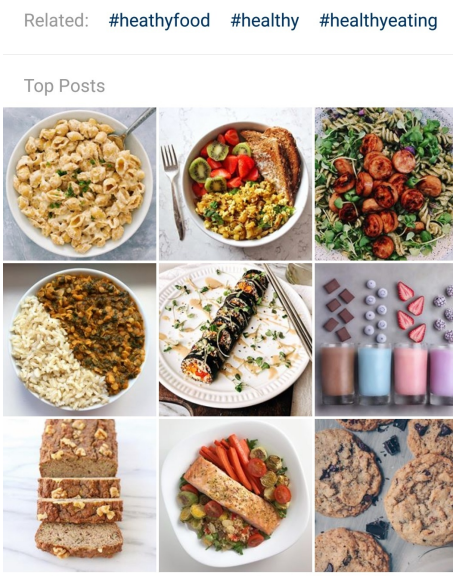


Fig. 1.5 Images uploaded to Instagram depicting food meals.

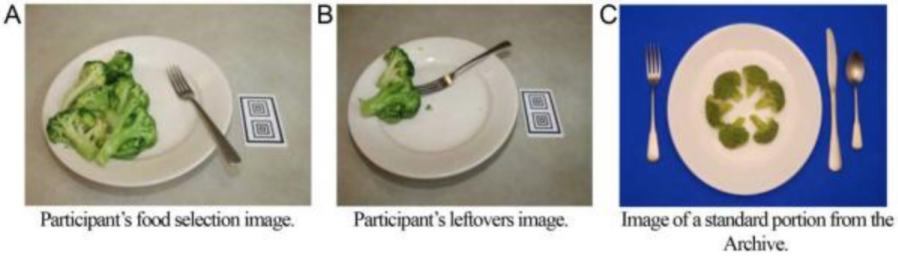


Fig. 1.6 Figure from [5] depicting meal images that are analysed using 'Food Photography Application' to determine nutritional intake.

Further research into the use of using digital photography to determine nutritional intake was conducted in [50] using camera worn sensors to personalise dietary intake. The aim of [50] was to assess dietary intake in sporting and the general population and to provide solutions to the problem of underestimation in dietary management using food logging. Figure 1.7 are image examples taken using Microsoft SenseCam from [50]. Forty-seven participants wore Microsoft SenseCam for one day to record dietary intake while also manually recording dietary intake using a food diary. Microsoft SenseCam recording and food diary were compared for analysis and to determine differences if any. For the manual food diary, participants were asked to describe the food items consumed such as time food

was consumed, brand name, and portion size. Microsoft SenseCam can provide a visual diary of what the participant consumed and was able to define portion size and leftovers for accurate nutritional analysis. Results from the analysis showed that using a food diary in conjunction with a visual diary can promote the accuracy of nutritional intake [50]. This research [50] highlights the importance of using images to provide additional information for accurate dietary management through providing insights into food leftovers, portion size, and food items that are not reported in food diaries [50]. It is clear that using images for dietary management has many advantages; however, human interaction is still needed to complete the process of image food logging, users still have to determine what the meal is and search nutritional information. Recent research has focused on determining nutritional intake from images using automated approaches to promote convenience and adherence to weight management applications.

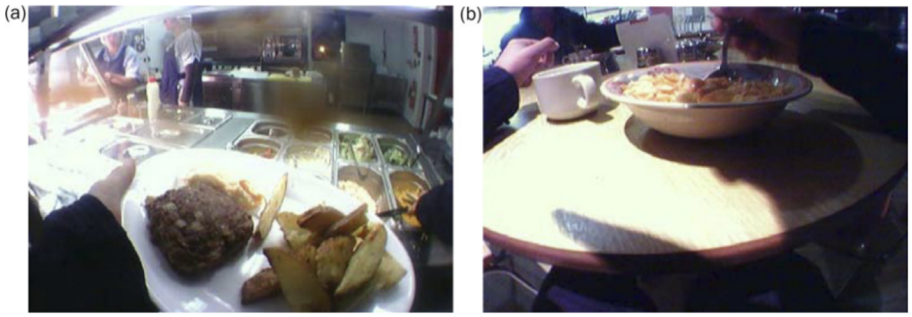


Fig. 1.7 Image depicting meal images taken using Microsoft SenseCam [50].

1.6 Automated Food Image Logging

Using food images for food logging also can allow for an automated approach. Automated approaches for food image logging involves predicting what food items are present in the food image and then searching the food item name on an online database to determine nutritional information. An automated approach may involve some user interaction in high-

lighting where the food item is located in the image or use image segmentation approaches to automatically isolate food portions to enhance classification accuracy. Computer vision approaches have been applied to achieve automated food image logging in various ways utilising image processing techniques. The conventional approach to achieve automated food logging follows a serious of steps. Visual features such as shape, colour, or texture are extracted from food image datasets. Relevant features are chosen that best describe food classes. Labelled features, which are extracted from a training image dataset, are then used to train a supervised machine learning classifier, e.g. Support Vector Machine (SVM). Once training is completed the classification model can be used to classify test images (features extracted from test images). For test images to be classified by a trained classifier, the same feature or feature combination need to be extracted. Figure 1.8 describes the image classification pipeline to achieve automated food logging for dietary management. In recent years large amounts of research have been published that explores the use of computer vision approaches in the classification of food images for dietary management [55,56]. The work presented in this thesis explores the use of computer vision and statistical approaches for food image logging to promote dietary management. Chapter 2 presents a literature review that discusses research that has used computer vision approaches for food image logging.

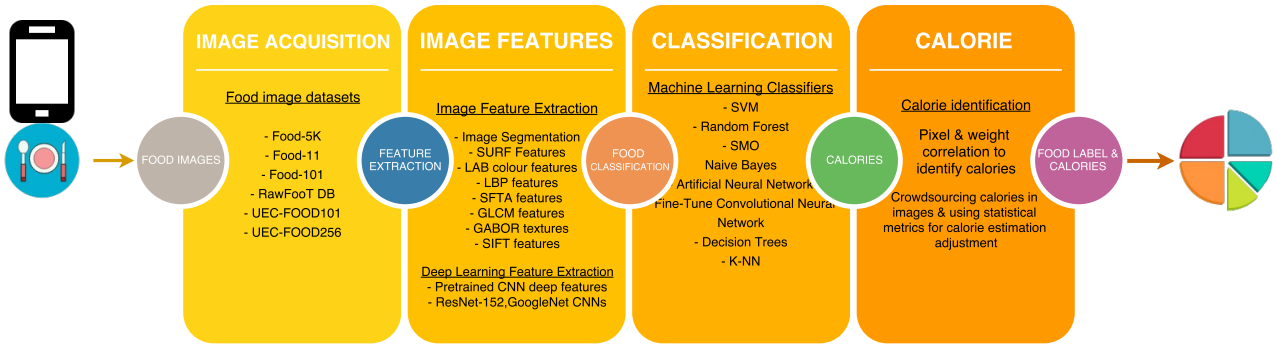


Fig. 1.8 Approaches that have been used in previous research for automated food image logging.

1.7 Research Focus

The focus of this thesis was to investigate, develop, and evaluate the performance computer vision and statistical methods that can be used for food image logging. Current conventional food logging interventions require high user engagement when documenting their energy intake. As discussed earlier, this may include searching online database APIs or taking note of the nutritional content of the food item to input into the food logging application. The aim of this research was to research computer vision methods to remove much of this complexity when compared to traditional food logging. The use of traditional feature extraction methods and deep convolutional neural network (CNN) features were investigated in detecting food images and specific food item classification and using statistical approaches with image processing techniques to determine calorie content in images. Crowdsourcing was also investigated to determine how the 'wisdom of the crowds' can be harnessed to provide dietary feedback using food images. The work presented in this thesis focused on using computer vision methods and statistical approaches to automate food logging through predicting food items and determining calorie content. The remainder of this section will discuss research Aim and Objectives along with research questions this research project seeks to answer.

1.7.1 Research Aim

The aim of this work is two fold: 1) to investigate, develop, and evaluate computer vision and deep learning approaches for food image classification and 2) To explore and evaluate methods that could be used for predicting nutritional content in food images.

1.7.2 Research Objectives

The following research objectives were highlighted to achieve the aim of this PhD project;

- To identify, develop, and evaluate methods to quantify calorie content of food items in pictures using image processing methods.
- To examine the use of crowdsourcing approaches to predict nutritional content of food images for dietary management.
- To examine the use of feature extraction approaches with supervised machine learning algorithms that could be used for food image classification to promote food logging.
- To examine the effects of using feature fusion for food **detection** in images.
- To examine the effects of using feature fusion to **predict** specific food items in photographs captured in free-living environments.
- To examine the effects of using feature fusion to predict texture images of food items.
- To examine the use of state-of-the-art pretrained deep learning models for deep feature extraction to classify food portions in images.

1.7.3 Research Questions

The following research questions were identified, which underpins this PhD project;

- What state-of-the-art food logging techniques are used for dietary management?
- How can food photography be used for dietary management in current literature?
- How can calories be calculated from a food portion in a photograph through correlating calories with pixels?

- How can crowdsourcing be utilised to support accurate calorie prediction to support dietary management?
- What computer vision and deep learning approaches are available for food image classification for automated food logging?
- How accurate are deep feature extraction approaches using pretrained CNNs (trained using ImageNet Largescale Visual Recognition Challenge (ILSVRC)Dataset compare to conventional feature extraction approaches in predicting food items in images?

1.8 Overview of Research Framework

Figure 1.8 is a proposed research pipeline that illustrates what technologies and methods are available highlighted in the literature review. The literature review was completed to highlight what technologies are available that could be used in each stage of food image logging. In this PhD research project, the use of images is a major component, and a preliminary literature review was conducted that researched image feature extraction types and calorie estimation methods. Figure 1.7 is a summary of automated food logging framework. This automated dietary management framework comprises of two areas; food classification and food calorie estimation. Each section in Figure 1.8 highlights different technologies and tools that can be used for food classification and calorie estimation. Figure 1.9 is a summary of the technologies that are used in a food image recognition pipeline.

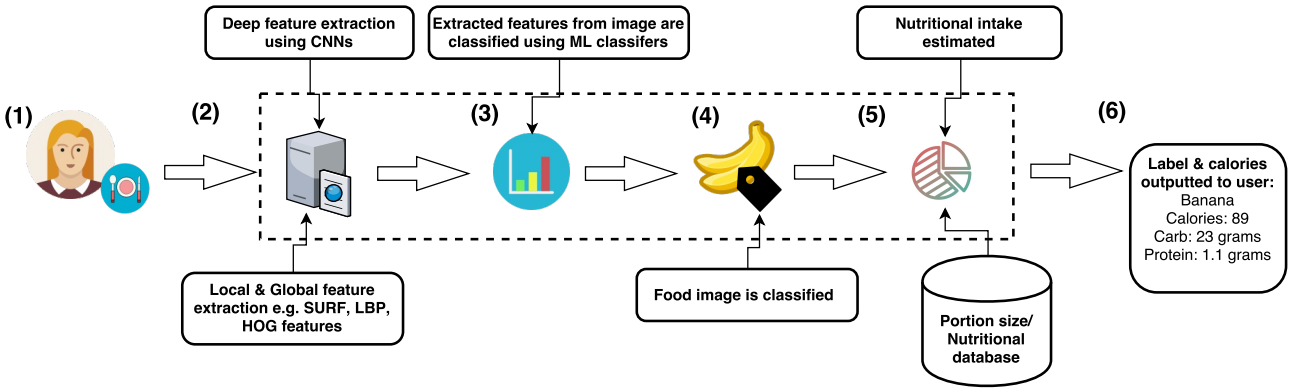


Fig. 1.9 High level system design to classify images and to estimate calories for food logging.

1.8.1 Research Study Design

This PhD research project uses an experimental design framework and consists of 4 studies. Each study presented in this work follows the following format; each study presents an introduction to the problem area and highlights related work and a high-level summary of what methods and technologies are currently available. Each study also presents research aim, objectives, and motivation. The rationale of each study is also highlighted in what the study seeks to achieve. Methodology section details what methods and technologies are used as well as evaluation methods. Results section documents what has been found using the methods presented in methodology section and a discussion section is presented after results. The conclusion section highlights key findings, research and clinical implications, and limitations. The remaining section will discuss research studies that have been proposed using the literature presented in Chapter 2. These research studies will presented as a work chapter based on the related publication listed in Chapter 1.

1.8.2 Study 1 - Semi-Automated System for Predicting Calories in Photographs of Meals

This Chapter discusses the use of image nutritional analysis techniques to ascertain a more accurate calorie reading from photographs of food items. The methods employed involve collecting a ground truth data set (pixels and calorie contents of each weighted portion) through correlating weight of a food item with its area in cm^2. This ground truth dataset was then used with a regression model to predict future calorie content of food portions. The proposed system uses a semi-automated approach to allow users to manually draw around the food portion using a polygonal tool, this would allow to user to accurately pinpoint the exact portion size. Related research also highlight the advantages of using a polygonal tool to pinpoint food portion borders for segmentation and accurate classification [8]. This study is exploratory in determining if pixel information can be correlated with weight and area for calorie estimation.

1.8.3 Study 2 - Automated Adjustment of Crowdsourced Calorie Estimations for Accurate Food Image Logging

The aim of this Chapter was to investigate the feasibility of crowdsourcing non-experts and experts in accurately determining calorie content in images of meals from an existing calorie estimation dataset. In previous research 2 participant groups were recruited; experts and non-experts. Participants were asked to complete an online survey to estimate calorie content of images. Analysis was completed on each group's calorie estimations using descriptive statistics. A calorie adjustment algorithm was also proposed using crowdsourced descriptive statistics. This Chapter is inspired by RFPM approach in which food images are analysed by remotely to provide nutritional content/ quality of food items.

1.8.4 Study 3 - Feature Fusion Food Image Classification to Support Dietary Management

In this Chapter, a feature fusion approach that combines a local and global feature types for classifying 3 distinct food image datasets. The aim of this study was to determine what feature combination and machine learning algorithm classifier is most efficient in classifying real-world food images and texture food images, and food detection. image feature extraction methods used in study 2 comprise of colour, texture, and Speeded-Up-Robust-Features (SURF) features. Research was completed that focuses on using various combinations of feature types and a range of machine learning classifiers. Three food image datasets were used to evaluate feature types and determine what combination is most efficient and with supervised machine learning classifiers.

1.8.5 Study 4 - Combining Deep Residual Features with Supervised Machine Learning Classifiers to Classify Food Image Datasets

In this Chapter, a number of research questions were determined; (1) how efficient are deep residual network features for detecting foods in images and classifying food datasets using conventional machine learning algorithms? And (2) how efficient are extracted GoogleNet deep features in predicting Food/Non-Food images and classifying images into high-level food groups in comparison to fine-tuned GoogleNet model? To address these research questions, a number of deep CNN architectures were reviewed that could be used for deep feature extraction. In this study, ResNet-152 CNN architecture was used to extract deep residual features to train machine learning classifiers to classify various food image datasets. This study describes how deep features were extracted from several food image datasets of different structure. ResNet-152 was benchmarked against GoogleNet architecture which is another well known CNN type. A number of food image datasets were used to evaluate

the performance of ResNet-152 deep features, Food-5K was used to train food detection models and Food-11 used to train food group classification models. RawFooT-DB was used to evaluate the use ResNet-152 deep features to classify food texture items with low-intra variance. Food-101 was also used to test to train specific food item classification models. Evaluation metrics were computed to measure the performance of each classification model trained for each dataset and results will be compared with authors of similar works.

1.9 Research Contribution

The following research contributions were achieved in this PhD project;

- Proposed dietary management system based on RFPM that combines deep learning approaches for food image classification and crowdsourcing for calorie estimation.
- Proposed a calorie calculation pipeline that combines user interaction for image segmentation and image processing methods with linear regression.
- Crowdsourcing calorie adjustment approach was proposed to promote accuracy in food logging for dietary management. Experiments completed that suggest that calorie crowdsourcing adjustment approach improves accuracy of food logging.
- Utilised feature extraction methods and feature fusion with Speeded-Up-Robust-Features (SURF), Segmented Fractal Texture Analysis (SFTA), LAB colour features, and local binary patterns (LBP) with supervised machine learning techniques to classify food image categories.
- Texture feature fusion was used (SFTA and LBP features) for image classification of isolated food image textures and achieves similar results to other state-of-the-art CNN deep feature extraction approaches.

- Combined CNN ResNet-152 deep feature extraction with machine learning classifiers to classify variety of food image datasets. Results achieves higher accuracy in determining food group types in comparison to other published works.

1.10 Publications

The following is a list of publications related to the research presented in this thesis;

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, "Combining deep residual network features with supervised machine learning algorithms to classify diverse food image datasets," Comput. Biol. Med., Feb. 2018., DOI:10.1016/j.compbiomed.2018.02.008., ISSN: 00104825. [Journal contribution]

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, "Semi-automated system for predicting calories in photographs of meals," in 2015 IEEE International Conference on Engineering, Technology and Innovation/ International Technology Management Conference, ICE/ITMC 2015, 2016. [International conference]

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, "A semi-automated food voting classification system: Combining user interaction and Support Vector Machines," in International Symposium on Technology and Society, Proceedings, 2016, vol. 2016–March. [International conference]

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, "Towards personalised training of machine learning algorithms for food image classification using a smartphone camera," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016, vol. 10069 LNCS, pp. 178–190. [International conference]

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, “A digital technology framework to optimise the self-management of obesity,” in Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct - UbiComp ’16, 2016, pp. 1126–1131. [Workshop contribution]

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, "Comparison of Machine Learning Algorithms in Classifying Segmented Photographs of Food for Food Logging", Proceeding of CERC 2016 Collaborative European Research Conference Cork Institute of Technology – Cork, Ireland 23 - 24 September 2016 www.cerc–conference.eu ISSN 2220 – 4164. [Regional conference]

- **P. McAllister**, A. Moorhead, R. Bond and H. Zheng, "Automated adjustment of crowdsourced calorie estimations for accurate food image logging," 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Kansas City, MO, 2017, pp. 1059-1066. doi: 10.1109/BIBM.2017.8217803 [Workshop contribution]

1.11 Thesis Outline

This thesis consists of 7 Chapters. The following section provides a brief outline of each Chapter;

- Chapter 1: Introduction

  This Chapter provides an introduction to the research area and challenges. This Chapter will also provide a background to this research area and the impact of obesity to society and health along with research aim and objectives informed by the literature review. This chapter also discusses the use of images for dietary management in regards to food logging. Research achievements and contributions are also provided along with publications.

- Chapter 2: Literature Review

  This Chapter explores current dietary management interventions to support individuals who are overweight or obese. A literature is performed that focuses on a wide range of digital interventions that has been researched and how these have been integrated into commercial applications. This section reviews novel approaches for automated food logging in regards image feature selection and machine learning models to promote food logging.

- Chapter 3: **Study 1** - Image Processing Methods for Determining Calories in Photographs of Food

  This chapter presents a study that focused on combining statistical analysis and image processing methods with user interaction to determine calorie content of a food portion in a photograph.

- Chapter 4: **Study 2** - Using Crowdsourcing to Determine Calorie Content in Images of Meals for Dietary Management

  This Chapter investigates the use of crowdsourcing calorie estimations for experts and non-experts and how statistical metrics generated from the crowdsourcing data can be used to correct future calorie estimations to promote accurate food logging.

- Chapter 5: **Study 3** - Feature Fusion Food Image Classification to Support Dietary Management

  This Chapter explores the use of feature extraction techniques in order to classify various food image datasets using a variety of supervised machine learning algorithms.

- Chapter 6: **Study 4** - Combining Deep Feature Activations with Conventional Machine Learning Algorithms for Food Logging

  This Chapter focuses on investigating the use of deep convolutional neural networks, in

particular deep residual networks features to classify a variety of food image datasets for dietary management.

- Chapter 7: Discussion, Future Work, & Conclusion

  This section discusses the main findings from each chapter and how each chapter can be combined to inform the development of a dietary management system. Future work is also discussed and potential studies are proposed to extend research work already completed in each chapter.

1.12 Summary

This chapter discussed the motivation for the research presented in this thesis. This chapter highlighted main research questions along with aim and objectives. This chapter introduced traditional food logging methods and more novel approaches for dietary management such as crowdsourcing nutritional content and automated food logging. This chapter also introduced how digital interventions and emerging technologies can be used in promoting dietary management through using images of meals for automated food logging. Research aim and objectives were also discussed in this chapter, which underpins the work presented in this thesis. Major contributions achieved and publications are stated along with a thesis outline, which gives a high-level overview of the contributions in each chapter and the approaches used.

Chapter 2

Literature Review

2.1 Introduction

Food logging is an essential method in the battle against obesity and this chapter explores different electronic food logging approaches for dietary management. This chapter also examines novel food logging approaches that utilise computer vision technologies and statistical methods for image classification and calorie identification. This chapter explores how smartphone technologies have been used to document food intake, and the increased use of smartphone applications have incorporated behavioural change methods to increase retention among users. This chapter gives an overview of what food image photography approaches have been used for food image logging and also what automated approaches have been explored in the literature. This chapter will detail what image processing and computer vision methods have been used in using images for the automation of food logging. Food images can infer greater detail in regards to food nutritional analysis and portion size, and this chapter explores what image processing and machine learning techniques have been used to achieve an automated approach for nutritional analysis through determining what food items are present in food images. The remainder of this chapter introduces smartphone

food logging techniques and computer vision methods and how these approaches have been applied for food image logging for dietary management.

2.2 Smartphones for Food Logging

The use of technology such as smartphones to monitor dietary intake is commonplace and this can due to the portability of the device as well as the availability of smartphone applications to search for nutritional content. Smart devices and applications can share user information, data, and progress to monitor dietary intake and weight to help with weight loss programmes. Research has been completed in the use of technology in regards to food logging to promote weight loss [33, 37, 43]. Research presented in [37] analysed the use a smartphone application 'Lose it!' along with a memo app on a smartphone, and also paper and pencil method for dietary management. The trial lasted 8 weeks, and 19 participants took part, and results show that app users recorded dietary information more consistently when compared to the paper and pencil group. Key take-home messages show that using smartphone applications and devices may have the ability to improve retention rates when monitoring dietary intake. Other research also compared using a smartphone application with a paper diary [28.] The aim of [28] was to determine the feasibility and acceptability of using a smartphone application as a self-monitoring weight loss intervention and to compare this intervention to a paper diary or website. One hundred and twenty-eight participants were randomised into a smartphone app, website, and paper diary group. The smartphone application was called my meal mate and incorporates behavioural change approaches through including goal settings, self-monitoring, and text messaging feedback. The trial lasted 6 months, and results show that there was a higher retention rate for smartphone application users (93%) compared to the website (55%), and paper diary group (53%). Results also show that adherence was statistically higher in the smartphone application group

compared to memo, and paper diary groups. Weight loss was also higher in the smartphone group compared to other groups [28]. The results presented in [28] highlight the potential benefits of using technology and smartphone for food logging to monitor nutritional intake. Similar research analysis highlights the relationship between application usage frequency and weight loss and results show that frequent food logging application usage is associated with improved weight loss. Further research is needed with larger group of participants and also to take into account participant characteristics. Research published in 2017 also discusses the impact of using smartphone technology in promoting dietary change in adults [5]. The objectives of the work presented in [55] focused on dietary modification to determine if a commercial health application can be used to promote dietary change. Participants were randomised into 2 groups; 1 group using a smartphone application to recorded nutritional intake and group 2 used a paper journal to estimate nutritional intake. Results show that nutritional intake reported by each group differed and the smartphone application group reported higher levels of satisfaction. Literature reviews were conducted that considered smartphone applications that try to encourage dietary self-regulatory techniques and strategies and challenges to promote weight loss for weight loss [56, 57]. Several studies were reviewed that highlighted intervention characteristics of smartphone applications and the studies highlighted in the literature review used various combinations, e.g. commercial food logging application, social media support, goal setting, and intervention groups. Conclusions and key findings of the review stated that the use of smartphone applications for intervention studies for dietary self-regulation should have a rigorous design process, reduce confounds and make the process easy to follow, adhere to reporting guidelines, analysis the statistical relevance of the effects of the intervention, and also to accurately assess the healthiness and nutritional intake of the food consumed by the participant.

Other similar research also explored the use of camera-based smartphone applications for food logging [58]. The purpose of this study was to evaluate an app that allows adolescent

users to document nutritional intake using images. Participants over 3-7 days and participants used the application to capture pre and post meal images using a fiducial marker for portion size context for more accurate dietary intake calculation. Results indicate that adolescents can use food image capture applications with ease to document nutritional intake. However, the research also highlights application usage retention rate as an area that needs more focus. Authors suggested that reminders should be incorporated to promote application usage. This research [38] support other research on the viability of using images to capture nutritional intake. The primary advantage of photo food logging, compared to traditional methods, it allows users to ascertain a large amount of information in regards to dietary intake. The following section will discuss approaches that utilise food photography for food logging.

2.3 Digital Food Photography

Digital photography has become a convenient tool for measuring nutritional intake. Using a picture to document dietary intake removes much of the complexity by not having the individual to manually use text input methods to search for dietary intake or inputting calorie amounts. To assess the efficacy and validity of using a digital photograph as a method to measure nutritional intake research in [59] was completed that explored the use of food pictures that could be used to estimate dietary content. The process of the Digital Food Photography Method used images to take photographs of user-selected food portions and their leftovers. Pictures of the same weighted food portions were used to compare with the user food images and leftovers. The photos were then sent to a laboratory for nutritional analysis by using trained raters with the use of an application 'Food Photography Application'. This application uses the images obtained during data collection, user food selection portion, leftovers, and the weighted food portion (ground truth) of the same food item to display to the trained rater. The trained rater compared the leftovers with the food

selected determined nutritional content by further examining the user food selection with the images of the weighted food portions. The trained rater determines what portion percentage the user selected using the weighted food portion image. Energy intake was calculated by comparing the percentage of the standard portion that the user selected and the leftovers. The Food Photography Application is then able to calculate food intake by analysing the content of food selection and the leftovers. This research highlighted the importance of visually documenting nutritional intake to give a more precise insight into the amount of food consumed and has potential to personalise food intake. Similar research also uses digital photography as an approach for dietary management. Research presented by Amougou et al. study the use of food photography to aid in determining portion sizes for dietary recall. Ground truth food portion images were prepared in the form of a food portion photograph book (FPPB) and were used by participants to recall the food portion size they consumed. Participants were also asked to self-serve food portions, and these food portions were also weighed. Twenty four hours after consuming the food portion, the participants were asked to select the food type and food portion using the FPPB. Results show that 77% of the portions were correctly estimated for adult group and 74% of food portions were correctly estimated for children [60]. Similarly work presented by Bouchoucha, et al. also researched the use of food image manual for determining portion size. A ground truth food portion dataset was compiled and validated by comparing with 24-hour recall method using images. Results from [61] indicate that using a 24-hour photo recall method is This research highlights the importance of food images in determining portion sizes for accurate dietary management.

2.4 Remote Food Photography Method (RFPM)

RFPM is a semi-automated approach in which the food images are taken on a camera-enabled device and uploaded to a server where the image is analysed to determine portion

size and nutritional intake by trained nutritionists [42]. Research has been undertaken to explore the use of RFPM. In some studies, RFPM method relies on the use of photographs of food items and the leftovers taken by the user within a free-living environment. The images are then sent to another location to be analysed for nutritional content. Participants examining the photos are asked to label the food items in the photos. Images can be analysed and compared against a ground truth dataset to estimate nutritional intake similar to [59] 'Food Photography Application'. RFPM combines automation with manual input to create a semi-automated dietary management framework, as ground-truth portion food size images are stored in a database for reference to calculate nutritional intake. In [62] research was conducted to investigate the reliability and effectiveness of RFPM, participants were instructed to send photographs of their meals, and clinicians would analyse them. The results of this study revealed RFPM was a reliable way to provide energy intake measurements to users. Other research [63] use algorithms to compare with other portions of food to ascertain nutritional content. Similar to [64], users send photographs of their meals to a server for analysis. Food estimation is then calculated by comparing user three images. Two images are users photographs, one photograph of users food portion and one picture showing leftovers. The last picture is a ground truth image used by the application to calculate energy intake. Results from this study [28] showed that RFPM overestimated food by 13%.

In [40] a combination of methods was employed, one that combines crowdsourcing and RPFM. This research enabled users to take photographs of their meals and send it to a crowdsourcing platform (Amazon Mechanical Turk). The photograph would then be analysed by several users working for Amazon Mechanical Turk) to estimate nutritional content. Results of this research state that this system overestimates calorie content by +7.4%, while dieticians overestimate by +5.5%. RPFM is another useful method that can be

employed to record nutritional intake, however, more research needs to be done in finding out ways to quantify food items for accurate food logging.

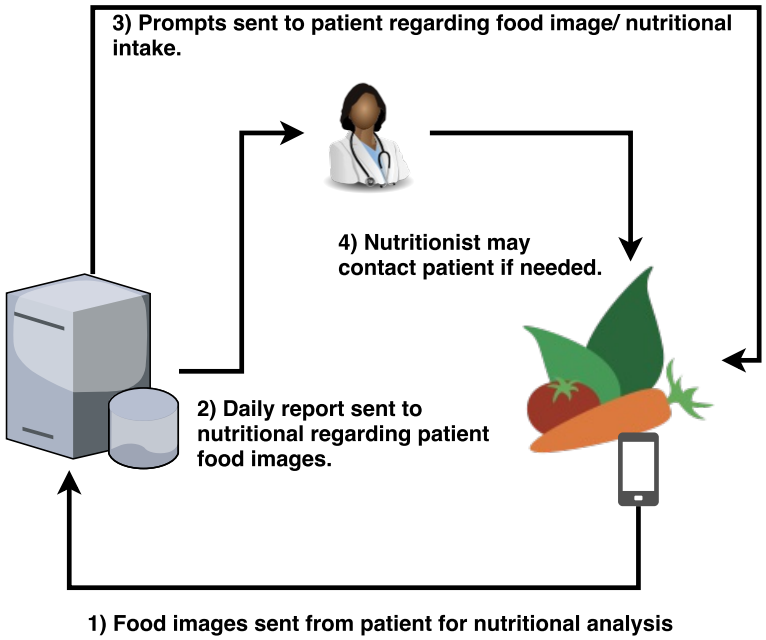


Fig. 2.1 RFPM for food image analysis for dietary management [7].

RFPM has been utilised to various problem areas, in [65] It was used as an exploratory method to accurately identify nutritional intake of infants to address the relationship between infants food intake and childhood obesity. The goal was to evaluate the nutritional content of simulated milk using RFPM compared to weighing the formula. Results show that using RFPM can achieve similar accuracy compared to the weighing method within 7.5% equivalence bounds among serving sizes and types of all servings. In other research, RFPM used trained image analyst to assess participant's weighed food image portions to determine weight and food categories and to determine the nutritional content [1] ultimately. This aim was to evaluate the viability of using RFPM as a means for providing dietary management [66]. Results show high correlation was reported between the digital method and the weighed records supplied by participants. RFPM highlights the importance of using images to ascertain more detailed information regarding weight and portion size to estimate accurate

nutritional intake ultimately. However, due to the nature of RFPM, dietary intake feedback in real-time will be difficult therefore there is a need to develop an automated or semi-automated approach. Computer vision methods have been researched to provide for food image classification and combining these classification techniques with nutritional databases and also image processing methods for calorie identification. Section 2.5 will discuss computer vision techniques that have been used in automatic food logging for dietary management.

2.5 Automated Food Image Logging

Automated food image logging is the process of automatically predicting food and nutritional content of a food portion within an image. As discussed, traditional ways of food logging such as using a food diary to manually document food intake can be tedious and time consuming, and this may affect food logging adherence [38]. The use of smartphone devices has grown, and dietary management applications are common in smartphone app stores. Current food logging methods may require individuals to manually input nutritional content from packaging or search online databases, and describe portion sizes, and this may lead to inaccuracies. Food logging adherence is also an issue for traditional and digital dietary management, and behaviour change theories have been incorporated to try and promote continual use through personalisation and reminders with some success [68, 69]. To address the difficulties of current food logging, an automated approach is needed that can address the issues of conventional food image logging. Food images have shown to be a convenient method of ascertaining detailed information regarding food type and portion size. Computer vision approaches have been applied to this problem area with varying degrees of success. To develop a food image recognition system, the most popular method has been to utilise a supervised learning framework, described in Figure 2.2.

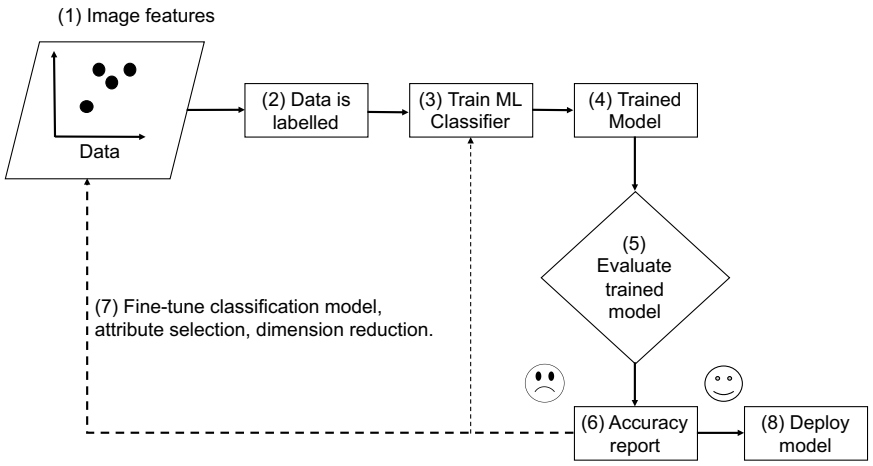


Fig. 2.2 Supervised machine learning process based on [72].

Most food image recognition systems utilise a supervised machine learning approach in which relevant features are extracted from images and these features are labelled and passed to a machine learning classifier, e.g. support vector machine (SVM), neural network, for training. Research has been completed in utilising different types of feature extraction methods on food images, and different feature extraction types have been used for various food image recognition problems, e.g. food detection, food type, and specific food item. Food image classification is a difficult task, due to high variance of foods within an image, food image quality (shadows, image obstructions), and multiple similar shape structures. Food portions captured in a free-living environment, e.g. restaurant, home, may contain obstacles or suffer from lighting conditions. Food images can also suffer from high-intraclass variance in which food images of the same class can look dissimilar, and food of different classes can look similar. Figure 2.3 is an example of a primary automated food logging application, in which the This Chapter will discuss 2 image classification methods that have been applied to food image classification, which are traditional computer vision feature extraction and deep learning methods. Figure 2.3 is a high-level overview of an automated food image logging system.

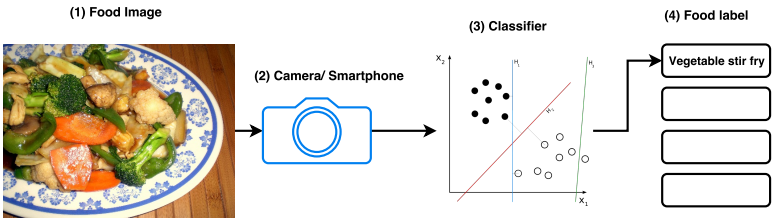


Fig. 2.3 Automated food image logging pipeline.

2.6 Computer Vision Approaches

The aim of computer vision to transform a captured image into a set of relevant features and to use these features to make a decision [71]. Feature extraction is a critical process in computer vision in which features are extracted to achieve visual understanding to accomplish a task. In computer vision, a feature can be described as an element of an image that is of interest such as colour, shape, edges, or texture [71]. These features can be extracted using a variety of feature detection algorithms once applied to an image at a specific area. Features can be categorised into 2 types; global or local features. Image global structures such as corners, blobs, edges, and lines can be extracted however prior human knowledge may be needed to understand what types of features need to be extracted from specific objects. A feature combination approach may be required to produce a feature vector that adequately describes the image or image area. Jiang et al. state that feature extraction can be broken down into types; human expert methods, image local structure-based methods, image global structure techniques, and machine learning based statistical approaches. Global and local features have been used extensively and applied to a myriad of classification problems. A global feature extraction is an approach that use features to describe the image in its entirety, and common global feature types are texture, histogram of oriented gradients (HOG), and colour.

Once visual features are extracted from an image they are then used with machine learning algorithms for classification. Supervised machine learning classifiers are widely used to train classification models for image classification. For this method, features are labelled and used with a machine learning algorithm for training. After training, the model can then be used to classify test instances. Various types of image features can be detected for food image classification such as corners, edges, texture, and shape and research has been completed in combining these feature types in enhancing the performance of food image machine learning classification accuracy. The remainder of this section will give an overview of popular feature extraction algorithms and discuss how these computer vision feature extraction approaches have been incorporated in food image detection and recognition systems [71, 72].

2.6.1 Scale Invariant Feature Transform (SIFT)

Scale Invariant Feature Transform (SIFT) is an algorithm which is able to determine and extract key features within an image [73]. The aim of SIFT is to successfully extract image features in an image independent of the following elements [73];

- Scale
- Rotation
- Viewpoint
- Illumination

SIFT features allow for a generation of feature sets that can provide for image matching and recognition regardless of scale or illumination. This makes SIFT a popular feature extraction method that can be employed in many problem areas. SIFT can compute a set of features that are not sensitive to the following constraints highlighted. Once the features

are calculated, they can then be used with machine learning algorithms for training. To compute SIFT features from an image a number of steps are completed. The first step is labelled 'Scale-Space Extrema Detection'. This stage utilises the blob detector algorithm DoG (Difference of Gaussian). DoG is used instead of LoG (Laplacian of Gaussian) due to its efficiency and LoG being too costly [73]. DoG applies a Gaussian blur using a pixel size called 'sigma' of different sizes on an image, and the resulting pixel intensities of both images are subtracted. Edges are not of interest when using SIFT features and DoG highlights edges as well as other interest points in an image. Low contrast interest points are removed, i.e. edge blob points are removed similar to using Harris key point detector. Once edges are removed from the generated interest points, the image will be analysed at different scales or octaves, to determine which scale an interest point is best generated. Once a strong interest point is determined, the octave (where that point was best represented) and interest point coordinates are saved. The next stage is orientation assignment for each interest point and this to ensure rotation invariance when matching points. An area around a key-point is taken, and the direction and magnitude are computed to determine key-point orientation [73]. After the orientation phase, a key-point descriptor is formed. This is achieved by taking an area around the key-point and broken down into sub-blocks. Orientation histograms are created for each sub-block to build the key-point descriptor. Lowe et al. has stated in [69] that the best way to match critical points is to use a k-nearest neighbour approach. A key-point can be measured against key points obtained from training images using k-nearest neighbour to obtain a match. The next section will discuss how SIFT has been used in identifying food items within obesity interventions.

2.6.2 Speeded-Up-Robust-Features (SURF)

SURF (Speeded up Robust Features) is an interest point detector that is also scale and rotational invariant. SURF features have been utilised in different literature to pinpoint

features in food items [74]. The main advantage of utilises SURF interest point detector over SIFT is due to speed. SURF is able to compute interest points much faster in comparison to SIFT. The main difference of SURF when comparing to SIFT is that SURF uses integral images, which leads to faster computation. Integral images use an approximation of the determinant of hessian blob detector [74]. This process can be calculated efficiently by using 3 operations as described in (2.1) [74]. SURF determines scale-space using a box filter and an advantage of using box filter as an approximation is that integral images can be used (2.2).

$$S(x,y) = \sum_{i=0}^{x} \sum_{j=0}^{y} I(i,j) \tag{2.1}$$

$$H(\mathbf{x},\sigma) = \begin{bmatrix} L_{xx}(\mathbf{x},\sigma) & L_{xy}(\mathbf{x},\sigma) \\ L_{xy}(\mathbf{x},\sigma) & L_{yy}(\mathbf{x},\sigma) \end{bmatrix} \tag{2.2}$$

$$\sigma = \text{current filter size} \times \left(\frac{\text{base filter scale}}{\text{base filter size}} \right) \tag{2.3}$$

SURF is able to pinpoint scale and location by using determinant of Hessian matrix (2.2). Where $H(\mathbf{x},\sigma)$ represents the Hessian matrix at scale σ in x. $L_{xx}(\mathbf{x},\sigma)$ represents the convolution of the second order derivative in image I at point x. The second order derivatives can be evaluated quickly and efficiently by using integral images [77]. The detector used in SIFT is based on Hessian matrix as it able to produce great accuracy. Determinant of Hessian matrix is able to detect interest points by expressing the change around a region in the image. For orientation assignment, wavelet responses in horizontal and vertical directions are used. The most dominant orientation is computed by using the sum of the responses within a sliding window. Feature descriptors are computed using horizontal and vertical wavelet direction, the neighbour around a keypoint and is divided into sub regions. Vertical and horizontal are extracted from each region and these are combined to determine a SURF

feature vector. SURF features have been successfully applied to various areas of detection, matching, and classification [75, 76].

2.6.3 Colour Feature Extraction

Colour features are common visually descriptives and are widely used in image recognition. Colour features are popular in regards to image retrieval and image indexing because of the simplicity of the process. There are several colour spaces that can be exploited and some colour spaces are much more suited to particular problem areas. This section will discuss the colour space types that can be used to extract a histogram of colour features for object detection and recognition.

2.6.4 RGB Colour Space

Red, Green, Blue (RGB) is a colour space that comprises all the colours that can be produced from red, green, and blue. RGB is one of the most common used colour specifications in representing images in computing. Each pixel within an RGB image comprises of three points (red, green, and blue) in order to represent another colour. RGB colour feature extraction has been used extensively in image classification and image retrieval within health informatics domain. In [79] RGB features were extracted from a food image dataset to train a number of machine learning models with the aim to develop a real-time food intake classification system. In order to extract RGB colour features, the authors of [1] extract a variety of RGB colour statistics by dividing the image into 16 x 16 window blocks and each 16 x 16 is then subdivided into 2 x 2 blocks . The mean and variance of each RGB block is extracted and the final feature vector is 24. The colour features described in [1] was used within a feature combination framework (along with Histogram of Oriented Gradients (HoG) and Local Binary Patterns (LBP) and the results from using just using RGB statistics to classify 18 category dataset was 37.89% [79]. Other research also utilise colour features in

food recognition [77] in conjunction with other feature types and using colour features alone achieved 63.76% accuracy across 15 food image types. It is clear that colour statistics play an important part in food image recognition as much research combine colour features with other hand-crafted features. From the literature it is clear that there is a major opportunity to extend the research in food image recognition in using colour spaces along with other different feature types.

2.6.5 LAB Colour Space

As well as RGB colour space, LAB colour space is also widely used in image recognition and retrieval. LAB colour space is described as a 3 axis colour system; L representing lightness and A and B representing colour dimensions [78, 80, 248]. There are several advantages to using LAB colour space as a method to represent colour in images; it provides a precise means of representing colour and LAB is device independent and also LAB colour space can easily be quantified to compare elements in images. [80]. LAB colour space is strongly related to human visual perception. The L channel represents different shades from light to dark, A channel represents red to green, and B represents yellow to blue. Research has utilised LAB colour space features to solve a wide variety of problems, in [81] RGB colour images were converted to LAB colour images in order to extract A and B channels values along with C and Y channels from CMYK colour space (from same image) in order to automatically detect diseased tomato plants. These colour features were used along with pixel values in disparity map and thermal intensity maps to train a SVM. Results from this work show that a combination of local and global features can improve the accuracy of detection of diseased plants [81]. Other research use LAB colour features for skin segmentation [82] and research also use LAB colour features for salient object detection in the wild with promising results [84]. Literature indicates the colour features are important in generalising between different objects.

2.6.6 HSV Colour Space

As well as LAB colour space, Hue, saturation, and value (HSV) is another 3D colour space that is able to quantify and describe colours based on their brightness and shade values. Hue represents the colour, saturation represents the amount of colour, and value is the brightness. HSV color space is commonly used in computer vision and robotic research due to being able to seperate colours based on intensity and is also commonly used for image segmentation. HSV colour features have been used extensively in object and scene recognition, object tracking [84, 85, 86], and also for image segmentation [89]. Equations (6), (7), and (8) describe how HSV colour space is computed [87, 248];

$$H = \begin{cases} 0^\circ & \Delta = 0 \\ 60^\circ \times \left(\frac{G\prime - B\prime}{\Delta}\right) & , C_{max} = R' \\ 60^\circ \times \left(\frac{B\prime - R\prime}{\Delta} + 2\right) & , C_{max} = G' \\ 60^\circ \times \left(\frac{G\prime - B\prime}{\Delta} + 4\right) & , C_{max} = B' \end{cases} \tag{6}$$

$$S = \begin{cases} 0 & , C_{max} = 0 \\ \frac{\Delta}{C_{max}} & , C_{max} \neq 0 \end{cases} \tag{7}$$

$$V = C_{max} \tag{8}$$

HSV colour space approaches have been used in a variety of problem areas for detection, classification, and content retrieval. In [84] HSV colour space was used to detect shadows in images and to suppress shadows for accurate feature extraction. Results from [84] show that shadow detection and suppression is able to improve object detection. In [88] Luminance

Chroma Blue Chroma Red (YCBCr) and HSV colour models were used to isolate faces by removing background with promising results. HSV colour space was also used to recognise partially occluded objects under different variations. HSV colour features along with RGB and YCBCr were used. Results in [88] show that HSV colour features achieved the highest Top-1 accuracy result with 40% in comparison to other colour spaces and a combination approach. Literature shows that there is potential in using HSV colour space in object detection, matching, and recognition. The literature also highlights that colour features are combined with other feature extraction approaches in order to enhance accuracy.

2.6.7 Histogram of Oriented Gradients (HOG)

HOG is a feature descriptor that has been applied to various object classification and detection problems. The descriptor is based on using dense grids of local intensity gradients and edge directions. To extract HOG features from images, the input image colour and gamma is normalised is first divided into a number of cells, and each cell contains a 1 dimensional of gradient directions over the pixels. The histogram for each cells are combined in order to compute a better representation. In order to enhance invariance to illumination, spatial blocks are normalised. The HOGs for all cells are accumulated to form a HOG descriptor [90].

Fig. 2.4 HOG cells computed to form HOG descriptor.

HOG descriptors have been used extensively, e.g. HOG have been used for human detection in images [90] and object recognition [91]. Figure 2.4 is an example of determining local HOG features in cells that can be used for object (food detection). HOG features have been applied to variety of areas such as finger print recognition, human detection, and emotion detection [90, 92, 93].

2.7 Texture Features

Texture features have been used extensively in content identification and retrieval based systems. Image texture features can be utilised in various ways in image processing such as texture segmentation, detecting shape structure, and texture classification. For texture classification, supervised learning is a popular method used in image classification by way of extracting texture textures from a group of defined sample classes. These features can then be used to train a machine learning classifier (e.g. SVM, Random Forest) to classify further texture features and texture features can be used to highlight distinguishing characteristics within a region of interest. This section will give a brief overview of texture feature extraction types and discuss texture feature representations that have been used in image classification and how they have been applied for automated food logging.

2.7.1 Structure Texture Feature Extraction

Structural texture analysis is concerned with the hierarchy of spatial arrangements within a region of interest through analysing local micro textures. Local texture elements and arrangements can be defined using placement rules by which texture shape and structural information can be extracted for classification or segmentation. Texture structural is able to provide a detailed description of the image and is more complex in comparison to statistical textural analysis. Using a texture structure approach reduces the texture feature extraction

problem down to determining a set of texture elements or 'texels' to desribe a repeating pattern in an image [94].

2.7.2 Statistical Texture Feature Extraction

Statistical approaches to texture feature extraction is concerned with determining statistical measures derived from colour or grayscale image intensities. Statistical approaches can be used for image classification and for image segmentation and texture statistical approaches differ in comparison to texture structure approaches as they do not attempt to determine image structural components or hierarchy but concerned with descriptive statistics properties to highlight patterns between gray level intensities [94]. Examples of texture statistical based feature extraction methods are Gray Level Co-occurence Matrix (GLCM), in which image properties are generated based on the relationship gray pixel occurence within a group of pixels and other statistical texture based measures also consider edge frequency within an image to determine the complexity of a region of interest. Fractal based texture analysis has also been used to measure textual geometry to describe spatial texture patterns. Fractal can be used to describe patterns within a region of interest at various scales and allows for statistical analysis of how complex geometric structure is in an image or region of interest.

2.7.3 Segmented Fractal Texture Analysis (SFTA)

SFTA is a feature extraction method that is able to extract texture information from an image [95]. The algorithm accepts an input image and the images are then decomposd into multiple binary images using a Two-Threshold Binary Decomposition (TTBD) method.

$$I_n(x,y) = \begin{cases} 1, & \text{if } t_l < I(\text{x,y}) \leq t_u. \\ 0, & \text{otherwise.} \end{cases} \tag{2.4}$$

where $I(x, y)$ is a set of binary images. Binary images are computed by using thresholds from T and using the Two-threshold segmentation as described in (2.3) [15]. t_l and t_u represent a pair of upper and lower thresholds. Pairs of thresholds are applied to the input image to obtain a set of binary images. The reason for applying pairs of thresholds to obtain binary images is to ensure that objects in the input images are segmented. The binary images that are outputted from the TTBD method can be described as a sub set of binary images that would have been outputted using a single threshold algorithm. SFTA feature vector is constructed using the binary images by extracting the pixel count (size), gray level and boundaries fractal dimension [15]. These measurements are used to describe object boundaries in each input image. The SFTA feature vector size is directly related to the number of binary images generated using the TTBD algorithm, for example if eight images were computed after using the TTBD algorithm on an input image, the SFTA feature vector would be 8×3 (3 being the number of measurements extracted from the binary images: size, fractal dimension, and mean gray level) [95].

2.7.4 Local Binary Patterns (LBP)

Local Binary Patterns (LBP) has its origins in 2D texture analysis and is based on the idea of comparing pixel information with neighbouring pixels. To create an LBP vector the following method is used, firstly, the area in question is divided into a number of the cells. The cells in the area are measured 3×3 pixels usually. The center pixel in the cell is compared with its neighbours. If the center pixels value is greater than the neighbour, then the neighbouring pixel is assigned as 0 or if the neighbouring pixel value is greater than the center pixel then it is assigned 1 as illustrated in Figure 2.4. After this process is completed, a binary sequence is then computed for each pixel within the cell. The binary sequence is computed to reveal an LBP code. A histogram is then generated to statistically measure the occurrence of LBP codes in an image [96, 97]. LBP have been used extensively

in image recognition and retrieval, in [98] LBP have been used in facial recognition for gender classification using real world images, and in [99] LBP patterns were used for facial recognition with different facial expressions with results achieving 97%.

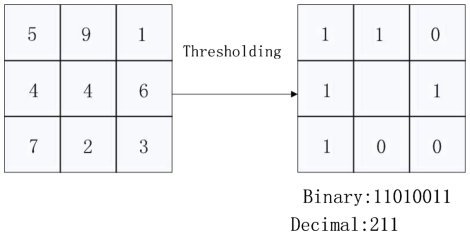


Fig. 2.5 Diagram describing how local binary patterns are computed (taken from [100]).

2.7.5 Gabor Filters

Gabor filters are used for texture analysis, feature extraction, and edge detection in images as they can be used band-pass filters. Gabor filters are orientation sensitive filters and are used for a variety of methods such as texture analysis, feature extraction, and edge detection. Gabor filters are band-pass filters and these filters enable certain filter frequencies and ignore other frequencies [101]. A gabor filter consists of a mask (or convolutional kernel) that convolves over an image input. This mask or filter convolves over each pixel in the image in order to produce a response and these responses are strongly represented when there is a sudden change of texture. Each pixel in the mask is given a weight and the mask convolves across an image. A change in response is then triggered between different texture region changes or edges in the image. There are certain parameters that can be edited in using a Gabor filter for texture analysis; the filter size (in pixels) can be changed and theta and sigma are also specified as well as Gamma parameter can also be altered which represents the spatial aspect ratio [101, 102]. Other parameters such as kernel type, lambda parameter, and phase offset. In regards to object detection and classification, Gabor filters have been extensively used texture detection and recognition. Gabor filters have been used in fingerprint

matching and identification [103]. Gabor filters have also used to extract texture features from a range of objects and scene classification [104]. Gabor filters have also been used for object segmentation [105].

2.7.6 Gray Level Co-Occurrence Matrix (GLCM)

GLCM can be used to extract extract second order statistical texture features and the values within GLCM matrix is able to describe the gray levels and brightness combinations within an image. GLCM is created by calculating how often a gray level pixel value occurs in a region compared to another pixel. Once a GLCM matrix has been computed, it can then be normalised in order to extract various texture features such as homoneity; which measures the homogeneity of an image, angualr second moment, contrast; local contrast variation within an image between a pixel point and the neighbourhood, Entropy; entropy measurements will be extracted based on homogeneity of an image, and Inverse Difference Moment (IDM) to name a few [106]. GLCM has been used to solve various problems such as MRI classification, tumour classification, texture segmentations, and biometrics [107, 108, 109].

2.8 Image Segmentation

Image segmentation is a process of isolated a region in an image for analysis. Image analysis using computer vision approaches it is common to only be interested in a certain region of an image therefore segmentation techniques may be employed. Image segmentation may be used to allow for more accurate image classification by remove noise or unrelated regions from an image to focus on relevant regions. Image segmentation is also able to enhance accuracy of recognition applications because once the region of interest is isolated and segmented, relevant features can be extracted and used with an object classification model.

Images can be divided into different parts if multiple regions need to be classified. This section will discuss image segmentation techniques that have been used in object detection and recognition.

2.8.1 Edge Detection

In image segmentation process, edge detection is an important stage that is able to highlight boundaries to allow for accurate segmentation. Edge detection is a common approach for determining changes in intensity values to highlight regions and shapes. Edges detection methods work on the basis of highlighting change gradient or brightness changes within an image. The set of points identified as an edge typically have a high contrast in comparison to other points within the local neighbourhood. The edge will then follow a continual boundary to ultimately highlight an object [110, 111]. There are various types of edge detectors available such as Canny method, Sobel Method, and colour segmentation. This section will discuss the edge detection segmentation methods that have been used for object identification and segmentation. The following section describes a variety of edge detection methods that have been used in image segmentation.

2.8.1.1 Sobel Method

The Sobel Method calculates the gradient for each pixel within an image in order to determine boundaries within an image. The Sobel Method is able to locate changes in spatial frequency in gradients by highlighting the greatest change to detect boundaries or edges. To determine gradient changes in pixels filters are used that convolve across the image. These filters are able to determine the y and x direction of the gradients. These filters are able to highlight changes relating to vertical (Gx) and horizontal (Gy) changes and the combination of these are able to determine the orientation and magnitude of the gradient using equation (2.4) [112].

-1	0	+1
-2	0	+2
-1	0	+1

Gx

+1	+2	+1
0	0	0
-1	-2	-1

Gy

Fig. 2.6 Convolutional kernels to detect edges vertically and horizontally.

$$|G| = |Gx| + |Gy| \tag{2.5}$$

Sobel method has been applied extensively in many areas. Sobel has been applied in bioinformatics for edge cell detection [113] and detecting edges of people in images [114].

2.8.1.2 Canny Method

The Canny Method is also robust edge detector that contains a series of processes in order to detect edges in an image input firstly, a Gaussian blur is applied to the image. The Gaussian blur is used to remove noise from the image in order to accurately highlight relevant edges. The second stage consists of a gradient operator that is able to compute the intensity of the gradient and the direction. Non-maximum suppression is applied after the gradient operator calculates the gradient and edge value, however the edge is still blurred. Non-maximum suppression is used to thin the edge through suppression of gradient values to 0 except for the local maximal. The final process is concerned with edge tracking to determine where edges begin and end, hysteresis is used by remove weak edges through an blob neighbourhood analysis. Weak edge pixels are analysed to determine if they are connected to strong edges, and if there is at least 1 strong edge pixel connected to the blob neighbourhood analysis then the weak edge pixel remains [110, 115, 116]. Figure 2.7 is an example of Canny edge detector being used on a food image. Canny edge detector has been applied to various areas such as scene segmentation, iris segmentation, and food segmentation [117].

Fig. 2.7 Example of using Canny edge detector method used to detect edges to be used for segmentation.

.

2.8.2 GrabCut Segmentation

GrabCut is an image segmentation method developed by Rother, et al. at Microsoft Research Cambridge, UK [118]. The motivation for Grabcut was to use graph cut methods to achieve foreground image segmentation with minimal user interaction. The user is able to draw a polygon around the region of interest to highlight the foreground and the image is then iteratively segmented. In [1] graph cut method was extended by using an iterative process to segment foreground regions from the background, simplify the process of segmentation with minimal user engagement, and also utilising the use of border matting for estimation of alpha matte around an object's boundary. For the initial segmentation process, the user is asked to draw a rectangle around the object of interest, by doing this the user is determining the background aswell as the foreground simultaneously. The object of interest is iteratively segmented to get the best result. In cases in which the GrabCut algorithm does not segment the object of interest accurately, a manual approach can be used the allows the user to highlight the areas that need marked as background. To determine the background a Gaussian Mixture Model (GMM) is used to determine the distribution of foreground and background pixels and a graph is then built from this distribution [118]. The edge pixels near to the foreground object are determined by comparing differences and colour and the image is

iteratively segmented to determine background and foreground. Grabcut has been employed for face segmentation, human segmentation, and video segmentation with promising results [119,120].

2.8.3 Colour Segmentation

Colour is another popular and efficient method for object segmentation due to the amount of information and intensities that can be processed from an image. There are different techniques available that utilise the use of colour to segment regions of interest from the image. LAB colour space has often been employed for image segmentation [122] as LAB colour space can be used to quantify and distinguish between different colours in an image visually. For image segmentation using LAB colour space, the first stage of the method is that a region of interest is selected from the area that is to be segmented, in order to analyse the layers in LAB colour space [121]. The brightness, chromaticity along the red-green axis, and chromaticity along the blue-yellow axis are calculated. The average LAB colour space is computed for the region of interest to compare with other pixels. In some methods, a nearest neighbour method, or K-means clustering is used to compare pixels with the computed colour markers for that region to highlight similar areas for segmentation [123]. Some methods use Euclidean distance to compare pixels with colour region markers to determine similarity if the distance between the two is small, that pixel would be classified as the colour region marker [249, 250].

Research presented in [251] use self-organising maps (SOM) with colour to segment multi-coloured fabrics. The paper offers a SOM approach to cluster colours together for segmentation; colour categories are initially located by dividing the U-matrix with an adaptive threshold. Colours with high similarity are then merged to decrease the possibility of over-segmentation. Results from [251] show that SOM colour segmentation can be efficiently used to segment separate colours in images. Other works using a supervised colour segmentation

based approaches to segment sky/cloud images is proposed [252]. Partial least squares (PLS) regression was used to analyse different colour spaces in images. The approach outlined in [252] does not need manually defined parameters as it is learning-based [252]. Other works also use supervised machine learning for colour segmentation, in [253] an extreme learning machine (ELM) was proposed to segment retinal vessel. Morphological features such as phase congruency, Hessian and divergence of the vector are extracted from each pixel from fundus images. These features are then used with associated labels as input into ELM classifier for retinal vascular segmentation. This proposed system shows promising results in segmenting retinal vessels with improvements in speed and robustness [253]. Thus research results and experimentation are promising in utilising colour statistics for object segmentation.

2.9 Supervised Machine Learning Algorithms

Once features have been extracted from images, the features are then used to train machine learning classifiers for image classification. A training set is defined and the chosen machine learning classifier is then trained using the training dataset. Supervised machine learning classification is widely used for image classification, however other machine learning systems can be employed in relation to the problem (unsupervised, semi-supervised, or reinforcement learning). In regards to supervised machine learning systems, data, or features, are already labelled in order to train the classifier. The remainder of this section will discuss popular supervised machine learning classifiers and examples of they are used in food image detection and classification.

2.9.1 Support Vector Machines

Support Vector Machines (SVM) are a popular set of supervised machine learning algorithms. SVMs can be classed and linear or non-linear classifiers. can integrate the use of non-linear boundaries to solve multiclass problems by using kernel methods such as RBF (Radial Basis Function), Polynomial, etc. These kernels are used to transform feature representation into a higher dimensional space to define hyperplanes to separate features from multiple classes [124]. SVM is on the most popular machine learning techniques and have performed well in generalising between a variety of classification problems such as food classification [125], face detection [127], and object detection [126]. As stated, in some multiclass problems, the data may become inseparable meaning that there is not a clear boundary definition. SVM parameters can be modified to enforce the use of a kernel to determine non-linear boundaries in a transformed feature space [3]. The hyperplane is considered optimal if a line is at the furthest distance from class data points (largest minimum distance) [127]. Kernels can be used that allow SVMs to solve multiclass classification problems, an Radial Basis Function (RBF) kernel is one of the most popular kernels used when training a SVM to solve multiclass problems due to the speed and can approximate non-linear function when tuning hyperparameters. When applied to a non-linear problem, the RBF kernel implements a margin around support vectors, and the gamma parameter is used to either expand the margin or limit the margin of each support vector to combine to determine classification areas in the dimension space. Through the tuning of the C (cost) and gamma parameter, the RBF kernel is very efficient in solving complex non-linear problems. Other kernels such as Polynomial and Sigmoid are also widely used in solving multiclass classification problems.

2.9.2 Sequential Minimal Optimisation (SMO)

SMO is a popular method to train support vector machines (SVM). Sequential Minimal Optimisation (SMO) addresses the quadratic optimization problem (QP) that result from using SVM to classify datasets. SMO is able to decompose the QP problem into smaller QP problems to be resolved using two Lagrange multipliers for each. The SMO algorithm is able to train a support vector machine using different kernels such as Polynomial Kernels, and Gaussian Kernels such as RBF [267]. Nominal attributes that can be used to label the data are transformed into binary representations and all independent attributes are normalised. SMO algorithm is able to analyse large datasets using the methods describe and performs faster in comparison to linear SVMs and non-linear SVMs [130]. SMO has been applied successfully applied in image classification and signal processing in a variety of domain areas e.g. biomedical informations [131] with promising results.

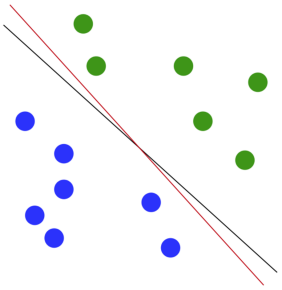


Fig. 2.8 Diagram depicting a simplistic example of using linear decision boundary to seperate training data. Important to compute the optimal hyperplane is seperate both classes in the dataset.

2.9.3 Artificial Neural Networks

Neural networks are influenced by the brain's functionality and are very popular in machine learning for solving complex problems. Artificial neural networks (ANN) can be applied to various problems such as image classification, regression, or clustering. Multilayer layer perceptron (MLP) is a popular type of ANN and is a feed forward neural network that

is made up of a number of layers. Each layer contains a number of nodes that are called neurons. The basic MLP architecture is made of three layers; input layer, hidden layer, and output layer and because of the amount of rich information/features it can be learned using a MLP it can be applied to problems that are of an non-linear nature. The basic function of a MLP is the ability to map features/ data into a set of outputs. Each neuron computes its input by using a weight that represents the strength between nodes. An activation function is then applied and there are a number of activation functions that are available i.e. sigmoid function, linear, and Gaussian. Once the activation function is applied, a single value is returned. Back propagation is used to train the MLP, the predicted output is compared to the expected output which is reflected in the cost function and the weights are altered. MLP training can be customised to suit the nature of the input dataset and problem, parameters such as training time (epochs), learning rate, and momentum can be configured [124, 131].

2.9.4 Decision Trees & Random Forest

Decision Trees can be used to solve both regression and classification problems and can be used with complex datasets. Decision trees utilise a tree structure in the sense that the further the tree extends the narrow the decision boundaries become and split into classes. The internal structure of a trained decision tree allows us to understand how decisions are made and provides opportunity for troubleshooting [132]. Figure 2.9 is an illustration of a decision tree taken from [213].

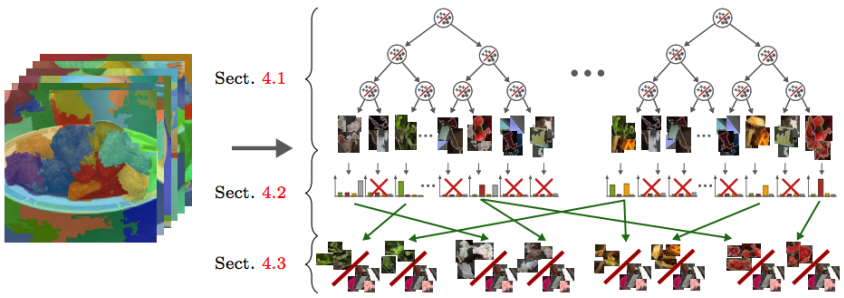


Fig. 2.9 Visualisation of a decision tree classification model.

Random Forest algorithm utilises a collection of decorrelated decision trees to train and classify a dataset [133]. This method is directly related to the bagging technique as the goal of the bagging technique is to develop a model with low variance and to average noise in the dataset. Random Forest algorithm is able to take subsets of the input data comprised of random values with each instance labelled with its class. For each subset created a decision tree is created, as depicted in equation 2.5 and 2.6.

$$D = \begin{bmatrix} i_{a1} & i_{b1} & i_{c1} & c_1 \\ i_{a2} & i_{b2} & i_{c2} & c_2 \\ i_{a3} & i_{b3} & i_{c3} & c_3 \end{bmatrix} \tag{2.6}$$

$$\begin{aligned} D_1 &= \begin{bmatrix} i_{a1} & i_{b1} & i_{c1} & c_1 \end{bmatrix} \\ D_2 &= \begin{bmatrix} i_{a2} & i_{b2} & i_{c2} & c_2 \end{bmatrix} \\ D_3 &= \begin{bmatrix} i_{a3} & i_{b3} & i_{c3} & c_3 \end{bmatrix} \end{aligned} \tag{2.7}$$

In 2.4, each decision tree D_n is trained using the subset training data and a classification for each instance is calculated. A majority voting rule is then used to decide on the final classification of the instance. Random Forest algorithm is efficient in that it is able to analyse large databases and is able to estimate missing data to help maintain accuracy [32]. In regards to food image prediction, random forests algorithm has been used extensively [133].

2.9.5 Naive Bayes

Naive Bayes is a set of popular machine learning algorithms known for their efficiency and minimal processing. They can be described as a set of simple probabilistic classifiers derived from Bayes Theorem. The reason the term naive is used to describe the algorithm

is because it assumes that attributes are independent of the associated class. This can be described in Figure 2.10 [134].

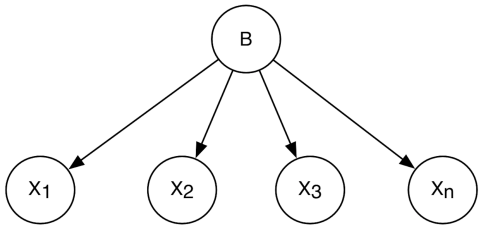


Fig. 2.10 Example of a Naive Bayesian classifier shown as a bayesian network. The predictive attributes (X_1, X_2, X_3,...X_n) are indepedent from their association class [134].

Bayes rule is enforced to compute the probability of a class based upon the values in the vector. Bayes' rule of conditional probability states that if you have a hypothesis H and the evidence (feature attributes) is connected to that hypothesis [5]. Naive Bayes assumes independence and the algorithm works efficiently and can outperform the most sophisticated machine learning algorithms on certain datasets. Bayes rule is shown in equation 2.5. Naive Bayes can be described as a simplistic approach to using learning probabilistic knowledge for classification. However, the present of redundant data can affect the performance and the introduction dependent attributes also diminish the performance of a classifier [134].

$$[h]Pr[H|E] = \frac{Pr[E|H]Pr[H]}{Pr[E]} \tag{2.8}$$

Naive Bayes classification has been used in determining activity logging [135] as well as food image classification for food logging [136], and text classification [137] with promising results.

2.10 Computer Vision Image Recognition Pipeline

Figure 2.12 is a conventional food image recognition pipeline and provides a summary of the various technologies that are available that could applied for automated food image logging. The pipeline details what food image datasets are available that are currently used for computer vision food image research. The next stage relates to image feature extraction methods that are used to train machine learning classification algorithms. Once the food image is classified then calorie estimation is applied and various methods are highlighted in 'Calorie' section of Figure 2.11 to provide personalised calorie estimation. The remaining section will discuss how these feature extraction methods and machine learning algorithms have been applied to achieve automatic food image classification for food logging.

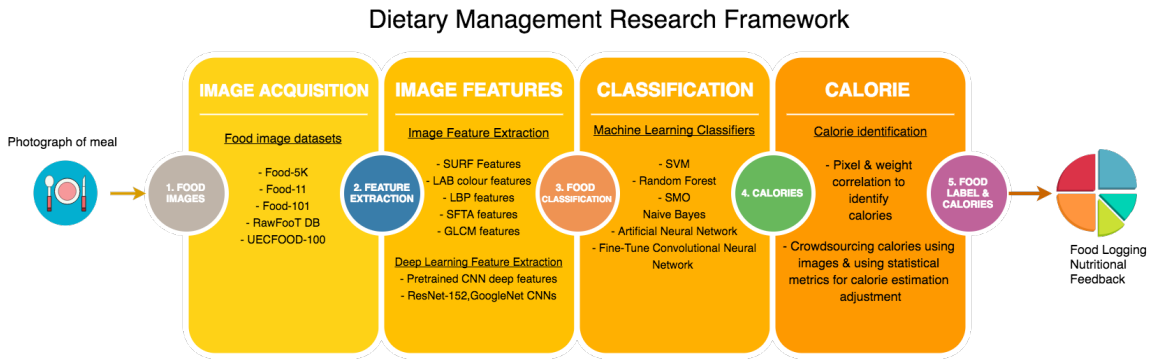


Fig. 2.11 Diagram highlighting computer vision methods that could be used for food image logging.

2.11 Computer Vision for Food Image Classification

There has been significant research within the area of utilising computer vision techniques for food image classification to support dietary management. Research has utilised various image processing approaches and feature extraction methods in order to identify foods and portion sizes for food logging. The exponential growth of smartphone usage and

camera built in functionality coupled with the increasing processing power of smartphones, the convenience of taking a photograph of a meal to determine calories to remove much of the complexity has been one of the main drivers behind much research. It is also important to mention of increasing processing power of personal computing that allow machine learning models to be trained more easily as well the use of graphic processing units (GPUs) to train deep learning models. The concept of using a smartphone to determine accurate nutritional content of food in an image as shown to be feasible and popular developers are using such technology and some commercial examples already exist on smartphone application stores.

This section will discuss the tools and algorithms available and also provide brief examples of how these methods been used to provide for dietary management in regards to food logging and activity logging.To develop an image classification model, the process is similar to that of training any other classification model. A set of robust features need to be determined and used to train a machine learning classifier. Features are extracted from an image dataset, and these features are used with a machine learning model. In much image classification research, models are trained using a supervised approach in which each set of features are labelled with the food item name. These labelled feature sets then train the model and a testing process follows to evaluate the trained model. Figure 2.4 outlines the basic pipeline to train a supervised machine learning classifier. The remainder of section will introduce computer vision and image processing methods and how they have been applied to support dietary management through automated food logging. Table 2.1 is a summary of various computer vision feature extraction approaches that have been used for food image classification.

2.11.1 Bag of Features (BoF) for Image Classification

Bag-of-features (BoF) or bag-of-visual words (BoVW) is a technique that is used to describe an image through a series of visual word occurrences using a visual dictionary.

BoF allows a feature vector to be generated through a feature selection process where by a vector is produced by completing various steps. Firstly, a feature extractor is applied to an image e.g. SURF, each patch extracted from an image is represented in a vector in the form of a descriptor. Feature reduction is then employed using K-means clustering in which similar features within the descriptors are clustered together. The number of clusters can be predefined and each cluster centroid represents a word (similar to that of a dictionary). Once the clusters have converged then each cluster centroid is used to form a visual dictionary. This visual dictionary can then be used to generate a feature vector for images. An image feature vector can be computed by counting the amount of visual word occurrences that are present in the visual dictionary. The results feature vector can then be quantified using a histogram to represent the number of visual word occurrences in an image. The feature vectors generated can then be used as input into machine learning algorithms. BoF has been used extensively to solve many image classification problems, malignant melanoma [138], video retrieval [139]. BoF has also been used to solve diverse problems ranging from classifying images of grapevine buds [140] to classifying food images for dietary management with promising results [141]. Figure 2.12 describes the BoF process used for image classification [277].

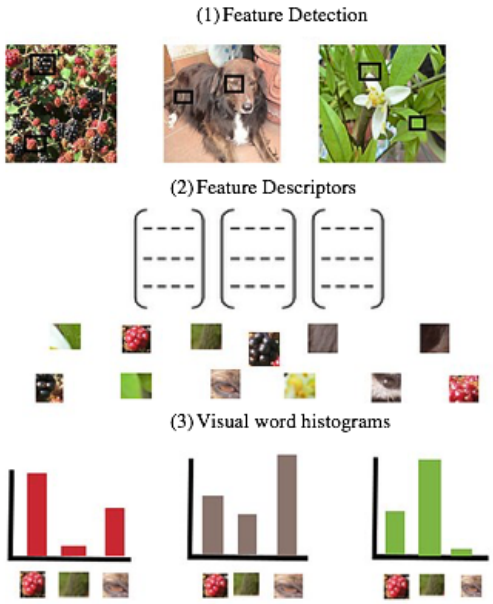


Fig. 2.12 Figure describing stages in BoF image classification method [277].

.

2.11.2 SURF & SIFT Features for Food Image Classification

Much research has utilised SIFT features as a way to extract hand-crafted features to use with supervised machine learning algorithms to identify food items in images. In [142] SIFT features was used to classify Indonesian food images, SIFT features were used to train K-dimensional tree and back propagation neural network to classify three Indonesian food types. The authors choose SIFT features for feature extraction due to scale invariance and partial invariance due to illumination, which is common in food image appearance. Results from [142] achieved 44% using neural network and 51% in using K-D tree. The research suggests that using SIFT features alone may not provide enough generalisation power and that a feature fusion approach may be needed to improve accuracy. Figure 2.13 is an example of SIFT features being extracted.

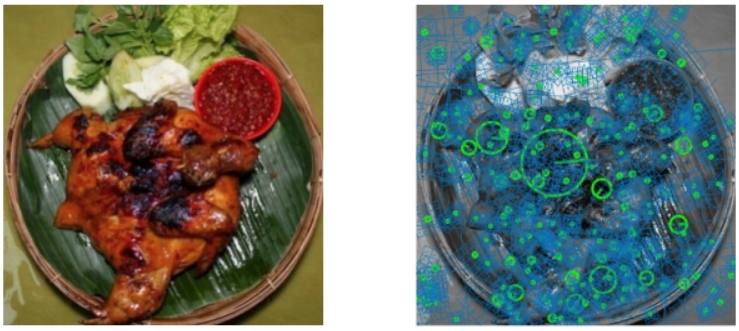


Fig. 2.13 SIFT features being detected in a 2-D grayscale input food image.

Other work compare SIFT features with SURF features in classifying UECFOOD-100, the authors of [143] combined SURF and SIFT features together to achieved 82.38% using a linear kernel with SVM classifier. In [72], SIFT was used to extract features to help classify food items and were coupled with colour histograms, and shape detection to increase accuracy. The system also updated the Bayesian model classifier based on user corrections of misidentified foods. Accuracy achieved by the system was 92%. In [145], SIFT was investigated in combination with colour descriptors. Accuracy rate combining different feature descriptors, including SIFT, was 82.8%. Other research [146], uses SIFT with Gabor filters in order to extract features from food images and then use these to 10 high level food groups. Results from this research show 55.8% accuracy rate. In [141] authors proposed an optimised BoF system that is able to provide dietary management support for individuals living with diabetes. Several different feature types are computed using the optimised BoF system, SIFT along with HSV features are used to determine a visual vocabulary and this vocabulary is able to compute a feature descriptor to use with a linear SVM. Results from this work achieved 78% in classifying high level food groups. Figure 2.14 is an example of using dense feature extraction as proposed in [141].

BoF is a popular method using for image classification and has been used for solve a variety of problems and has been used extensively with SURF and SIFT for food image

Fig. 2.14 SIFT features in HSV colour detection from [141].

classification [144, 147]. BoF has also been used extensively for food image classification for dietary management in regards to food image classification. Research presented in [148] used BoF used classify PFID (Pittsburge Fast-Food Image Dataset) (one of the most widely used food image datasets for computer vision research). The image dataset contains food images from fast food chains and the images were taken in a controlled laboratory setting and for the experiments images were grouped into into high-level food catergories such as bagels, pizza, salads, and sandwiches etc. The authors used BoF with SIFT and BoF with colour histograms. Three-fold cross validation was used to evaluate both methods and results were promising with BoF with colour highest performance was achieving 0.74 classification accuracy with sandwiches group 0.66 with salads. BoF with SIFT achieved 0.90 classification accuracy for salads group and 0.69 for donuts, desserts, and snacks group. Other related research [4] proposed a model that used colorSIFT features with BoF. Results from [141] showed that colorSIFT features used with BoF achieved 78% accuracy classifying 11 food catergories using a linear SVM. Further research [149] was completed that used PFID dataset to compute features using Maximum Response filter bank. Filters were used at difference scales and orientations to achieve scale invariance. Four dimensional feature vectors produced from using maximum response filters were passed to BoF model for feature selection to generate the vocabulary. Images were encoded using the BoF to train an SVM. Results from [149] show that texture features with BoF achieved 67.9% accuracy using 3-fold cross validation in classifying 7 high-level food groups, as proposed in [149].

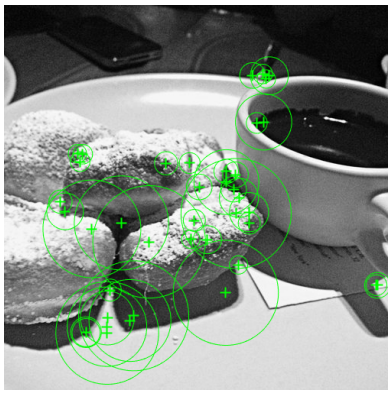


Fig. 2.15 SURF features detected in the 2-D grayscale input food image using detection option.

Research discussed illustrate the flexibility of using BoF in classifying food images for dietary management, however more research needs completed in using feature fusion with BoF encoded image features with other global features in classifying food image datasets taken in real-world environments (instead of laboratory based environments). BoF has proved successful in classifying food images however are certain disadvantages; the curse of dimensionality is an issues in regards to overfitting. To use BoF a visual vocabulary size needs to be defined to describe images and usually k-means clustering is used to cluster together similar feature points after initial feature extraction. Therefore the visual vocabularly can compute a large feature descriptors which can overfit. Research has been completing in using visual vocabularies of various sizes to try and determine optimal feature descriptor size with varying results. To overcome the issue of large dimensionality and scalability issues, feature reduction methods could be used as discussed in [150]. The literature indicates that BoF particularly used with SIFT and SURF features have experienced some success in food image classification for dietary management.

2.11.3 Colour Features in Food Image Classification

Colour is an important feature type in object classification and has been used extensively in food image classification. In [151] colour features were used to classify UECFOOD-

100 dataset, the image is divided into overlapping windows of 16x16 and each window is subdivided in 2x2 blocks. For mean and variance for each RGB channels was extracted for each block and Principal Component Analysis (PCA) is applied to the final descriptor. UECFOOD-100 was used in [151] and with just using RGB colour features, authors achieved 37.85%. Also in [151] using colour with other features such as LBP and HOG features increase the accuracy considerably with 53.35% accuracy. In [152] colour features were used for food recognition [152] in conjunction with other feature types and using colour features alone achieved 63.76% accuracy across 15 food image types. It is clear that colour statistics play an important part in food image recognition as much research combine colour features with other hand-crafted features. From the literature it is clear that there is a major opportunity to extend the research in food image recognition in using colour spaces along with other different feature type. In [152] using colour features combined with other feature types, the accuracy increases to 90.41% using texture, shape, and size features. Also in [3] colour histograms with BoF to for food image recognition, the authors in [149] incorporate the use of bounding boxes in order to enhance food recognition. In regards to extract colour features to train a machine learning classifier for food recognition, the authors in [149] divided the image into 3 blocks and extracted 64 bin RGB histograms from each block. Combining RGB colour histogram with BoF-SURF features achieved top-1 accuracy of 42% in classifying 100 food image categories.

In [153] BoF was used with RGB colour feature points, to compute RGB colour features 3x3 patches from images to extract local features. Results of using bag of colour features using patch method achieved 80% in classifying 10 food image types. Further research in [4] showed that using BoF with colour features with BoF and texture features increase the accuracy considerably. Other work [154] utilise colour features for fruit classification, colour features were extracted by images being discretised into 64 bins and the pixel number for every bin is counted. Colour features were used with morphology based measures

(MBM) and Unser's texture measure (UTM), authors provide no results of individual colour RGB feature results, however with the combined approached of the features the authors used, an accuracy of 89.5% was achieved in classifying 18 fruit image types using a single layer neural network and 10-fold cross validation. From research discussed it is clear the colour information is a very important aspect for food image classification, especially when compared with other feature extraction methods.

2.11.4 Texture Features in Food Image Classification

In recent works for food image classification, texture features have also played an important role in enhancing classification accuracy in regards to food image classification. Texture features have been used extensively for feature fusion with shape and colour features. In [155] authors combine the use of local texture information and global features for food image classification. LBP features along with spatial distribution of texture structures to determine shape context is combined determine food images using PFID. LBP and SVM were compared with SIFT and SVM and results show in [155] that LBP and SVM slightly outperformed SIFT features, further experiments combining shape features with LBP show that classification performance generally improved in comparison to SVMs trained with single feature types. The work presented in [155] also suggest combining further features together to improve performance. Other work presented in [156] used texture features to classify food images for food logging carbohydrate intake to support individuals living with type-1 diabetes. The work in [156] combines image segmentation with texture features to ultimately classify the food portion. LBP were used with LAB based colour features to train a non-linear RBF SVM to classify food image vectors and authors achieved 87% accuracy using 10-fold cross validation across 6 food groups. Authors noted that food image types with distinct colours achieved high results however food image types that share similar characteristics such as pasta, potatoes, and rice experienced several misclassifications. Research used texture

features derived from gray level co-occurrence matrix (GLCM) with HSV colour features to classify fruits in packaging and authors achieve 91% in recognising segmented fruit items in packaging [157]. Also in [158] GLCM was also used to generate texture features to classify food images, 10 GLCM features are computed Angular Second Moment (ASM), Contrast, Variance, Correlation, Inverse Difference Moment (IDM), Entropy, Sum Entropy, Sum Variance, Cluster Shade, and Maximum Probability. SIFT and GLCM features were compared in classifying 4 image types and results show that GLCM outperforms SIFT with BoF, however due to the small nature of the dataset used in [158], a larger dataset is needed to evaluate the effectiveness of GLCM based texture features. Research in [158] used GLCM based text features to classify sugar bulk foods using an ANN, authors used a number of texture features extracted from 10 types of sugary food objects. Results from [159] achieved 90% in using GLCM features with ANN in classifying sugar foods, and authors of [1] suggest combining colour features with texture features to enhance performance. The work described in [159] contains only 10 categories, more research is needed in using GLCM texture features with food items that contain low-intra variance to obtain a better indication of performance.

Research also completed in [160] proposed a feature descriptor that combines information relating to colour and texture information called Color Intensity Local Mapped Pattern (CILMP) from several resolutions. Texture information is extracted using sampled local map pattern (S-LMP) and for colour information, features are extracted based on the magnitude of RGB colour vectors within a 3x3 pattern map. To evaluate this texture colour descriptor, RawFooT-DB dataset was used, which contains 68 food texture types photographed in variation lighting conditions and authors in [160] achieved an average accuracy of 95.59% and improves upon previous methods used on the same dataset. The authors suggest that researching other descriptor methods that are invariant to lighting changes. This work could also be improved by incorporating other local feature methods along with the proposed colour texture information, i.e. SIFT or SURF. From the literature, it can be stated the textual information

does play an important role in food image classification, however more research is needed to using GLCM, and LBP features in classifying large texture datasets with low-intra variance.

Texture features, colour features, and global shape structure features were combined for fruit recognition in [161]. PCA feature reduction method was used to reduce the size the combined features structure. Unser's texture measurements (mean, contrast,homogeneity, energy, variance, correlation, and entropy) were used and combined with colour features and MBM perimeter, area, convex area, Euler number, solidity, major length, minor length, and eccentricity. PCA was used with these combined features and used with a single-hidden layer feed forward neural network. Further work in [161] could expand the fruit dataset and incorporate images of fruit in different locations and also use other machine learning classifiers to compare results with neural network presented in [161]. Experiments results were promising in achieving 89.5% accuracy using the methods highlighted in [161]. Other research also used statistical information based on colour and texture distributions along with a BoF model to predict food items in images. LBP are used to extract texture features from patches of the food images as well as pixels. RGB colour features are also used results from [162] are promising in combinig colour and texture feature types for food image classification. Work in [162] could be improved and extended by using other features e.g. SURF or SFTA based features to enhance accuracy with other features used.

2.11.5 HOG Features in Food Image Classification

In [163] RootHOG features, inspired by RootSIFT [6], were used with deep convolutional features, and RGB colour features to classify images of food. RootHOG feature is computed through an element-wise square root of the L1 normalized HOG features. Initially in [163], HOG features are extracted using 2x2 blocks and within each block eight orientations to form a histogram. Once the HOG patch has been computed, RootHOG

is extracted. Authors in [163] encode RootHOG patches into Fisher Vector (FV) to form RootHOG-FV and results show that using RootHOG-FV with one-vs-rest linear classifier and 5-fold cross validation achieved 50.14% classifying 100-food type image dataset. Also in [150] HOG patch descriptor encoded into a fisher vector is used to classify 100 class food image dataset. HOG features are extracted by dividing a local patch into 2x2 blocks and extract 8 orientations from each block, similar to work in [150]. For each patch 32 dimension feature vector is computed which is then normalised using L2 normalisation. PCA is also applied to the 32-dim vector to reduce to 24-dim feature vector. In [164] authors propose a mobile based food recognition application without user intervention, the system is able to detect the food region and crop the area, and also extract features for classification. A number of low level features were extracted to train food image classification model including HOG features, dense HOG features were extracted with a grid step of 8 pixels. At each point in the grid 31 HOG features are computed, 4 blocks of HOG descriptors based in 2x2 neighbourhood were combined to produce a 124-dim descriptor. HOG features are encoded into Fisher Vector along with dense SIFT features. Five-fold cross validation was used to evaluate and train a multiclass linear SVM with stochastic gradient descent. Results using dense HOG features achieved reasonable accuracy results with the highest being 85.7% for classifying 'cheerios' cereal. Feature fusion was used in combining SIFT and dense HOG features (of varying different window sizes used with each image) as proposed [165] to classify 85 food items with promising results. It is clear from the published research that HOG features can be successfully applied to food image classification for automated food logging. Table 2.1 summarises feature extraction approaches that have been utilised for food image classification. Table 2.1 is inspired by work presented in [276] and extends the table with other works.

Table 2.1 Conventional feature extraction for food image classification [276].

Author	**Feature type(s)**	**Approach**	**Food Dataset**	**Accuracy**
Hoashi et al. [165]	Colour, texture, HOG	MKL[1]	Food-85	62.5%
Matsuda et al. [146]	HOG, SIFT, Gabor, colour	SVM-MKL	UECFOOD-100	21.0%
Kawano et al. [163]	RootHOG	SVM, k-NN	UECFOOD-100	50.14%
Kawano et al. [166]	HOG, Colour	SVM	Own dataset	49.7%
Liu et al. [167]	Colour, HOG	Fisher Vector	UEC-Food100	59.6%
Yanai [168]	Colour, HOG	Fisher Vector	UEC-Food100	65.3%
Yang et al. [169]	PLF[2]	SVM	PFID	78.0%
Kawano [170]	GrubCut SURF, Colour	SVM	Own dataset	81.55% (Top-5)
Beijbom et al. [171]	SIFT, Colour, LBP, HOG, MR8	SVM	Menu-Match	51.2% (MAP)
Bossard et al. [213]	LAB colour & SURF	Random Forests	Food-101	50.8%
Anthimopoulos et al. [141]	Colour & SIFT	BoW & SVM	Diabetes	78.0%
Anthimopoulos et al. [141]	Colour & SIFT	SVM	Own dataset	87.0%
Wu et al. [173]	SIFT	Sift-Matching	Own dataset	<70.0%
Hariadi[174]	SURF	Euclidian Distance	Own dataset	92%
Pham and Thi Thanh [175]	SURF, Colour, Shape	Bhattacharyya Coefficient	Own dataset	86.39%[3]
Dalakleidi et al. [176]	SURF, Colour, LBP	SVM, k-NN	Own dataset	94.2%
Ciocca et al. [177]	RGB, Gabor, OG, LLC, CM, LBP, CEDD	k-NN, SVM	UNIMIB2016	0.759 (RGB HIST + SVM)
Zheng et al. [178]	SIFT, Colour, Improved Fisher Vector (IFV), LDC[4]	Linear - SVM	PFID, UEC-FOOD100	50.45%, 70.84%

[1]Multiple Kernel Learning (MKL)
[2]Pairwise local features (PLF)
[3]Precision result
[4]Linear Distance Coding

2.11.6 Image Segmentation in Food Image Classification

Research has been completed in automating food segmentation using various methods [172, 179-183], in [179] a dietary management system was described that incorporated food segmentation with food image classification. Food image segmentation was completed using local variation described in [180], which is based on graph based image segmentation and is able to measure similarity and differences between pixels. Local variation segmentation is able to measure the variability within a neighbourhood to perform segmentation. In [181] image segmentation was achieved using RGB fruit images through computing a high contrast gray value image based on RGB components of original image. Global thresholds are computed in order to determine background and foregrounds for segmentation, and finally to use morphological operations to fill in holes in the segmented image. Results using this method showed AUC score of 0.99 in segmenting 45 images. Graph Cut segmentation has also been successful in segmenting food images, in [182] Graph Cut segmentation was used to obtain the best contour boundary of objects, texture segmentation, and colour segmentation is also used with Graph Cuts method. Results show that using a combination of different segmentation methods is able to improve accuracy of food classification. However more research needs completed in segmenting foods in real world scenarios instead of lab based food images. A semi-automated approach could be incorporated that allows the user to highlight certain sections of the food item to guide the segmentation process.

In other works [183], canny edge detector was used to determine initial regions to be used for segmentation. Canny operator was used to determine strong edges to remove region of interest from background image. Canny edge filter was also used to determine regions with potential food items [183]. In [184] a semi-automated approach was suggested that used user input to determining food regions to allow for food segmentation. Research in [184] also presented a method for dish detection using Canny edge detector, the food image containing the dish was downsized and its grey-level level was equalised, afterwards the Canny filter

was used to detect edges of the dish. For food item segmentation, 2 methods were compared automatic region growing and semi-automatic. For automatic food item segmentation, seeds were placed within the elliptic region, which is the dish. Seeds are generated inside the dish region and seeds grown into full regions using colour features. For semi-automatic food item detection, the user is able to manually swipe the individual food items, the regions grow similar as the automatic method. Results show that the semi-automated approach was more accurate in segmenting food regions. The research presented in [184] suggests that minimal human interaction to guide segmentation may be beneficial and also reduce processing time. Sobel approaches have also been utilised for food image segmentation, in [278] a Sobel region-growing based approach was proposed where edges were detected using Sobel approach to merge regions together for food segmentation. Recent research also successfully employed Sobel edge detector to determine edges in fruit images. Authors of [278] stated that Sobel was selected for edge detection as it is able to performs better at noise suppression and image smoothing. Sobel approach was then used to isolate the fruit portion to allow for accurate size feature extraction [278]. Similar works also employ Sobel approach for segmentation of more complex food types such as pizza, pork, and potatoes [279]. Research indicates that using edge detection approaches such as Sobel and Canny can be successfully employed to segment a variety of food images.

2.12 Calorie Estimation

2.12.1 Crowdsourcing

Crowdsourcing can be described as a distributed problem solving method that allows multiple individuals to provide input or interactions. Crowdsourcing can also be described as an open call for individuals to take part in an internet based collective activity. Other definitions exist as a way to address problems by issuing task to a group of users [185, 186].

Crowdsourcing has been applied to solve various problems through collective participation, in [187] crowdsourcing was used to issue tasks to individuals and micro-payments are issued to users upon completion, in [188] online calls were open to allow users to submit ideas, suggestions, or solutions to issues. Also in [190] web tasks were issued to a crowd of individuals and were paid upon completion and in crowdsourced was proposed [189] in order to help businesses/ enterprises during the life cycle of a product or service. From these examples, it is clear that crowdsourcing can be used in various ways to solve problems through collective contribution. Wikipedia is a platform that uses crowdsourcing elements such as collective individual contribution to achieve a clear goal through the use of the internet. Amazon Mechanical Turk is also an online platform that allows 'workers' to complete tasks for organisations and receive micropayments for completing. Amazon Mechanical Turk has been used in research for dietary management in determining nutritional information in food images using crowdsourcing [40] and also for transcribing spoken language [280]. The following section will discuss research that use crowdsourcing elements to provide dietary management through analysing food images.

2.12.2 Crowdsourcing for Calorie Estimation

Crowdsourcing uses 'wisdom of the crowds' to allow a group of individuals to complete an activity to reach a goal or to solve a problem [185, 186]. This technique has been applied to dietary management in investigating how individuals perceive the nutritional content of foods. Research has combined RPFM and crowdsourcing to determine the food type, food size, and calorie content in an image through using Amazon Mechanical Turk [40]. In [40], tasks were repeatedly completed by Amazon Mechanical Turk workers to provide a nutritional workflow for dietary management. Results from these experiments indicate that using crowdsourcing to determine the nutritional value of meals is nearly as accurate as trained dieticians, however, each worker is paid and may take a significant amount of

time to complete the entire process, therefore a cheaper more efficient method is needed in using crowdsourcing for dietary management. Research completed by Johnson et al. propose a system called FoodFinder that allows crowds of people to estimate the weight of food meals [191]. A dietician was also used to estimate food weights and also to compare and contrast the relationship with crowds of people. Results show that a crowd of 5 individuals underestimated true meal weight by 63 grams and the experts (dietician) overestimated meal weight by 28 grams [191]. FoodFinder state that their system cost £3.35 for a crowd of 5 individuals and it took 2 minutes and 55 seconds for each image. This is cheaper than the methods proposed in [191] using Amazon Mechanical Turk, however, an automated approach is needed to provide for quicker dietary management that utilises previous measurements and matches these measurements to similar food items images. A computer vision approach could be incorporated that automatically determines food type and portion and connects similar portion sizes to nutritional information already determined through crowdsourcing.

Smart Food was proposed by Moorhead et al. that also compared experts (dietician) and non-experts estimation performance using food images in an online survey [192]. Participants were asked to estimate the number of calories (Kcal) or kilojoules (KJ) in meal images captured using a smartphone. Results were analysed and showed that the amount of calories difference between actual calories and estimated calories for non-experts was +55% (SD 79.9) and for experts +8% (SD 15.1). Results also show that that mode estimate from the crowd of experts is more accurate than 79% of individual experts. Other findings indicate that non-experts average median was more accurate than 63% of individual non-experts. Key messages from this research indicate that using crowdsourcing and food image logging for calorie estimation may be more accurate than most individuals estimating calories [192]. Each group (experts and non-experts) only contains 12 participants, to improve this study the number of participants could be increased for each group for greater analysis. In similar works combining images and food image logging, crowdsourcing was used for dietary rating

of food images [193]. In [193] a healthiness scale was used that allowed users to rate each image and results show there was a high correlation between user ratings and indicate that crowdsourcing can be used for dietary feedback. Similar research used a traffic light diet approach to assess the nutritional quality of images [194]. Results show that the ratings achieved high accuracy (>75%) when examining all foods and that there is promise in utilising crowdsourcing for dietary feedback [194]. Other research used crowdsourcing to analyse menus of restaurant chain food menus to determine healthiness of food items [195]. However, more research and investigation is needed in how individuals perceive food portions and types in a free-living environment as individuals may underestimate or overestimate nutritional content in certain meal types. Crowdsourcing can also be a useful tool to gauge individuals relationship with food items and in other related works, the use of crowdsourcing to investigate dietary preferences across different demographics, location, and time was examined [196]. Research from [197] uses crowdsourced data in a "big data" style approach, and results validate previous studies by correlating food preferences with specific populations and also public health patterns [197]. Thus the literature indicates that crowdsourcing may be a useful technique for food logging to promote dietary management.

2.12.3 Calorie Estimation Using Computer Vision and Image Processing Approaches

In regards to traditional nutritional logging, several studies have used smartphones as a way for users to recall nutritional intake [226, 227] by adding diary entries using a food database. In [226] a smartphone application uses behavioural strategies by using goal setting and self-monitoring and to then provide feedback. This is a continuous process to establish a feedback loop. The user can log energy intake (foods and drinks) to promote healthy living. Another study [29] uses healthy eating index ratings based on dietary guidelines with a scoring system. This index system breaks food down into different components such

as dairy, vegetables, and protein, etc. Within the smartphone application, the user can add meal portions according to key components for quick entries [29]. Other research present a mHealth application that utilises text messages, notifications, and NFC scanners [228]. Users can scan radio-frequency identification (RFID) tags to attain the nutritional content of food items [25]. Another study uses a smartphone application to act as a 'wellness diary' [229]. The wellness diary allows users to input daily observations and weight measurements as a form of health management.

Current research in image nutritional analysis show ways to identify and attain calorie content of different food items. In [77], the aim is to improve the accuracy of segmenting food items in an image using colour k-mean clustering. Colour k-mean clustering is used to create regions of similar colour to help classify items. For each item, segmented feature extraction is then performed to ascertain what the food item is. Features extraction such as size, colour, shape, and texture is also incorporated. After each portion is identified the calorie content is identified through nutritional tables [77]. Other related research [230] employ similar techniques to identify food items, using colour shape, and texture. In [231], food is identified by locating contours to pinpoint boundaries of each item. The application is implemented on a mobile device and once the image of the food is analysed the application requires user input to select the generated food item suggestion. Regarding user input, there are applications that are semi-automated which allow users to manually draw a boundary box around the food to segment food items for analysis [170]. Other research use reference objects next to the food to attain the correct size context. This process is used to calculate the portion size along with other feature selections [231]. There has been research completed relating to crowdsourcing to estimate calories within food portions.

Other research presented in [232] use computer vision based approaches to estimate the carbohydrates in food images for individuals with type-1 diabetes. Results from [232] show that computer vision based approaches were able to estimate carbohydrates with an

error rate below 20 grams. The system described in [232] combines automatic segmentation with classification to determine the carbohydrates in portions. Colour and texture features were used for classification and colour segmentation is used to segment food portions. The work described states that the food must not overlap or food items may be biased, however in future work an option for a semi-automated approach may be incorporated that allows the user to manually segment the food item using a polygonal tool. In other works, depth sensors were used, and authors report that the use of depth sensors for nutritional estimation enhances perform of previous work proposed by the same authors [233].

2.13 Deep Learning Approaches for Dietary Management

Deep learning based machine learning algorithms utilise elements previously discussed in regards to ANN. Deep learning approaches have been used to solve complex problems, especially for image classification. The popularity of deep learning for image classification can be attributed to superior performance in solving complex problems and has been applied to computer vision, natural language processing, speech recognition, and signal classification. In regards to classification, deep learning can highlight prominent features in data input to allow for accurate classification. There are many different types of deep learning methods available, i.e. deep neural networks, deep belief networks, convolutional neural networks (CNN), region-based CNN, or recurrent neural networks [72, 198, 199]. CNN has become very popular in regards to image classification which is evident in ImageNet ILSVRC competition in which recent winners have utilised CNN approaches of deep learning [198, 199]. The remainder of this Chapter will discuss CNN based methods for image classification and how they have been applied to promote dietary management.

2.13.1 Convolutional Neural Networks (CNN)

The popularity of CNNs has increased dramatically with them being used by researchers to tackle a broad range of image and voice recognition problems [21]. The use of pretrained implementations of CNNs gives the potential for applying them to a variety of problem areas without having to train a CNN from scratch. Convolution is used to describe the type of neural network as the input image is broken down into smaller overlapping shapes to determine certain patterns in the image. This takes place in the 'convolutional layer', and these overlapping segments are called filters. The patterns detected, by each overlapping shape in the filter, may consist of colour contrast or certain interest points such as edges. The overlapping shapes look for the same pattern on the image. The overlapping tiles are effectively used as input for a small neural network. This is done for each tile in the image. Each network in the filter holds the same weights to determine interest points in each tile. The output of this process is an array which each section corresponds to the network that describes patterns in each tile. A down-sampling process is then triggered after the convolution stage, and this is typically completed using max pooling where the representation divided into non-overlapping rectangles. Within each region, the maximum is retained. This process can be repeated many times to create deeper and more detailed representations. Fully connected layers are also present with a CNN architecture and are connected to activations from the layer previous. The fully connected layer takes the input from previous layers and uses this for classification using a soft-max function. Backpropagation is typically used to train the CNN in which the forward propagation is used to determine the error and gradient descent is then used to update the weights and parameters based on this error. This is repeated to train the CNN using a training dataset [9,22][72, 198, 199]. Figure 2.16 depicts the basic architecture of a CNN as proposed by LeCun [201]. Figure 2.16 is a CNN architechture proposed by LeCun in 1998 called 'LeNet-5' [201]. LeNet is a popular CNN model and is

widely used for classifying images of hand-written digits. LeNet-5 is a simple CNN that comprises of 2 convolutional layers and 2 subsampling layers, and fully connected layers.

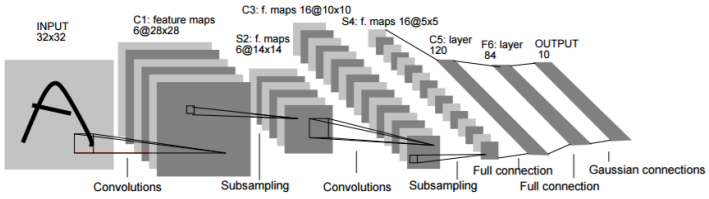


Fig. 2.16 CNN architechture depicting different layers [26].

There are various ways in which CNNs can be used for image classification;

- Training a CNN

  Much research has been dedicated in building CNN from scratch by specifying layers to create an architecture to solve particular issues. This process involves deciding on a combination of layers and using back-propagation to train the network. The architecture of a CNN may be different depending on the number of layers included and this depends on the problem and data that you are using. Research has shown that many convolutional layers can be implemented in an architecture interconnected with ReLu and max-pooling layers. Other architectures using a softmax layer for categorical classification or a regression layer for continuous responses [72].

- Transfer Learning or Fine-Tune Pretrained CNN

  Transfer learning is used when a trained CNN is applied to a new problem by retraining the final layers of the CNN. The last few layers can be fine-tuned for a new image classification task. Using transfer learning or fine-tuning to train a pretrained CNN can reduce training time as you are only training the final few layers by using the features in fixed layers to classify the new categories in the dataset [72].

- Deep Feature Extraction

  Instead of training a CNN from scratch or fine-tuning a CNN, pretrained CNNs can

be used for deep feature extraction. An image can be inputted into a pretrained CNN and layers in the architecture will generate responses and activations. Early layers in the CNN will generate primitive features such as edges and deeper in the CNN architecture more complex features are generated by combining features from previous layers. Deep feature extraction exploits this by inserting an image and extracting deep features from layers. These deep features can be used to train conventional machine learning classifiers such as SVM, Random Forest, or even a forward-feed neural network. Main advantages are that deep feature extraction can be more time efficient in comparison to fine-tuning or training a CNN from scratch, able to achieve similar accuracies (dependant on the dataset), and also no GPU or high-end hardware is needed for deep feature extraction [202].

2.13.2 CNN Architectures

Below is a list of popular CNN architechures that have been applied to tackle a range of image classification problems. These architectures have been used for fine-tuning and deep feature extraction for a wide variety of tasks. This section will discuss these popular architectures and how they have been applied for food image classification.

2.13.3 AlexNet

Alex Krizhevsky developed the AlexNet CNN architecture won the 2012 ImageNet ILSVRC challenge achieving 17.5% top-5 error rate. AlexNet architecture utilised convolutional layer stacking instead of stacking a convolutional layer and a pooling layer. The architecture is significantly larger than LeNet architecture (1998). Regularisation techniques were used in AlexNet architecture to reduce overfitting, drop-out is built into the architecture and data augmentation is used with the training images by flipping the images randomly.

Local response normalisation is also implemented to allow for feature map specialisation to improve the generalisation power of the model between classes. Figure 2.17 shows internal layers of AlexNet [72, 203].

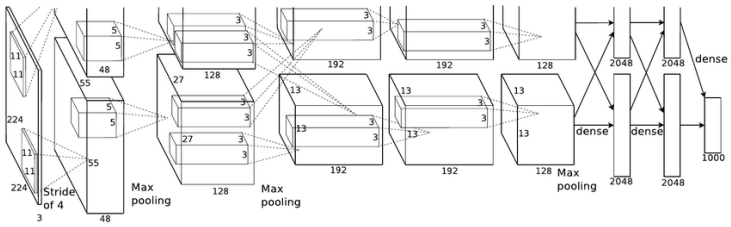


Fig. 2.17 AlexNet CNN architechture taken from [203].

2.13.4 GoogLeNet

GoogLeNet was developed by Christian Szegedy from Google Research and the architecture won the ImageNet ILSVRC 2014. GoogLeNet model achieved a top-5 error rate of 6.67% as the architecture is larger than previous trained models and the incorporation of inception modules to reduce the amount of parameters needed to train the model. The motivation for GoogLeNet was that larger CNNs may encounter the problem of overfitting as there is a large number of parameters used in the network. GoogLeNet's main contribution is the introduction of Inception modules that utilises the concept of using approximation of sparse structure with repeated dense components. Dimensionality reduction is used in order to ensure computational complexity is kept to a minimum and to avoid bottlenecks in the architecture. For comparison, AlexNet roughly contains 60 million parameters while GoogLeNet contains around 6 million. In regards to the Inception module in the GoogLeNet architecture, multiple convolutional layers are used to capture information at different and these layers use a ReLu activation function. Each layer also uses the same stride of 1 and the same padding, this is to ensure efficient stack combination of the convolutional layer outputs at the concatenation layer. The aim of the inception modules is to capture more detailed information utilising three convolutional layers contained in each. The Inception modules

can be describe as small CNNS with the larger CNN architechture, in which a variety of convolution size filters (1x1, 3x3, 5x5) as well as a pooling layer are used with the output of the previous layer. When constructing a CNN, a decision has to be made in deciding what size of filter to use within a convolutional layer, the authors of CNN suggest using a variety of different sizes. The output of these filters are concatenated Figure 2.18 is an example of Inception module used in the GoogLeNet architecture where a variety of different convolutional filter sizes are used and concatenated to be used with the next layer. [72, 204].

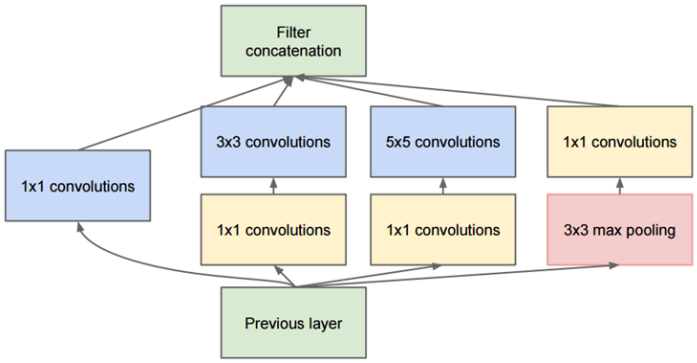


Fig. 2.18 Inception module used in GoogLeNet architecture [204].

2.13.5 VGG

Work completed by Karen Simonyan, et al. [205] explored the use of increasing weight layers in CNN to improve performance. VGG model achieves higher accuracy in comparison to previous architectures before it (AlexNet), and instead of using large convolutional filters, multiple 3x3 convolutional filters are used that convolve at every pixel using stride 1. The advantage of using multiple stack convolutional layers of 3x3 would allow the use of 3 non-linear retification layers to allow the decision function to be more discriminative. Also another advantage is that less parameters are computed, the combination of multiple stacked small convolutional filters allow for more complex features to be extracted due to the use of multiple non-linear layers that increase the depth of the network [72, 205].

2.13.6 Residual CNN (ResNet)

ResNet-152 is a deep residual pretrained CNN [206]. At the time of development, the authors of this CNN have described it as 'the deepest network ever presented on ImageNet' (2015) and is based on utilising 'extremely deep nets' with a depth of up to 152 layers. A residual learning framework which allows training of networks easier to converge and promote increased accuracy. The main advantages that residual networks contribute is the acceleration of speed in training networks, the effect of the vanishing gradient problem is reduced, and increasing the depth of the network which results in less parameters. ResNet-152 is made up of residual connections that allow important information to be transferred between layers. Residual connections allow a gradient to pass backwards directly through layers without losing vital information, in a regular CNN, the gradient must always pass through an activation layer [206]. This can cause the gradient to diminish, to circumvent this problem, connections within a CNN are appended with a shortcut that allows gradients to pass through thus decreasing the effects of vanishing gradient (information loss). Experiments using residual connects (ResNet-152) have reported increased accuracy and lower training times, in comparison to other state of the arts. The authors of ResNet-152 compare their work with other established CNNs and state that this residual deep net is 8x deeper than VGG nets [206]. Figure 2.19 is an example of a residual connection used in ResNet architecture [72, 206].

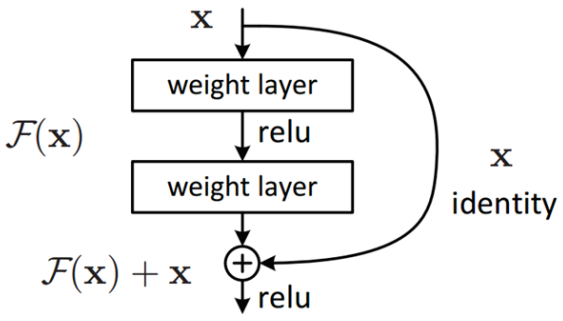


Fig. 2.19 Residual connection used in ResNet CNN architecture [206].

2.13.7 Region Based CNN (R-CNN)

The aim of RCNNS is to enhance the accuracy of objection detection using CNNs by using a bounding box over the identified object. R-CNNs attempt to tackle the problem of identifying multiple objects in an input image. The authors of R-CNN recognised that images may have multiple objects in them and therefore region proposals must be incorporated to highlight multiple potential objects in a single input image [207, 208]. In [207] R-CNN architechure us proposed that used region based proposals with CNNs and in order to determine multiple regions in an image. To determine multiple regions a 'selection search' method was used. Selection search is able to determine objects in images based of similar colour or texture and once objects have been identified then the region proposals containing each object are cropped and deep CNN features are extracted. These CNN features are then classified using a SVM to determine object item. Once the object inside each region proposal is classified using the SVM, the bounding box is then improved using linear regression model. Linear regression model is used to improve the coordinates of the bounding boxes. The resulting image is the original input image overlayed with bounding boxes for each object in the image and the label attached. The work in [207] incorporated region proposal method and classified the objects in each proposal using a pretrained AlexNet. The proposed R-CNN is described in Figure 2.20 taken from [207].

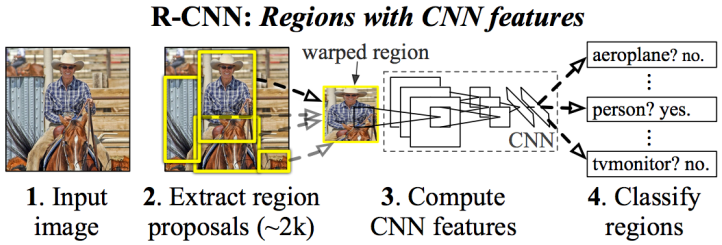


Fig. 2.20 R-CNN architechure incorporating search selection for region proposals and SVM for object classification taken from [207].

2.14 CNNs for Food Image Classification

2.14.1 CNN for Food Image Detection

In recent years, CNNs flavour of deep learning has been used in much research for dietary management with promising results. In regards to the food image classification pipeline, it is important first to determine if food is present in an image. Once food has been detected in an image another machine learning classifier can be used to determine the type of food. CNN has been utilised to determine if food is present within an image to provide for dietary management. This problem can be condensed down to a simple binary classification problem (food/non-food). The purpose of food image detection process is first to determine if food is present in an image or video. In regards to a food image recognition pipeline, this stage would be the first stage in food image recognition framework, i.e. determining if the picture is of a food image or non-food image. CNNs has been used to detect food in an image, in [209] GoogLeNet pretrained model was fine-tuned using Food-5K dataset. The training process in [209] utilised a subset of Food-5K data using 1000 iterations. The learning rate was 0.01, and the learning rate policy was polynomial. Results from [209] achieved 99.2% accuracy in determining food/non-food classes. Other research also utilised CNNs for food detection [210] and used 6-fold cross validation with different hyper-parameters to determine optimal settings and experiments achieved 93.8% in food/non-food detection.

2.14.2 CNN for Food Image Recognition

Extensive research has been carried out in utilising CNN for food item recognition. The food item recognition process would take place after the food detection phase in which the food item is predicted within the food image. In [211] CNNs were utilised to extract features from convolutional layers to determine if an image contains a food item and experiments

achieved 70.13% for 61 class dataset and 94.01% for 7 class datasets, these experiments used AlexNet deep features with a Support Vector Machine (SVM) with PFID dataset [211]. In [212] the aim of the work was to compare conventional feature extraction methods with CNN extraction methods utilising UECFOOD-100 dataset. Results from [212] achieved 72.6% accuracy for top-1 accuracy and 92% for top-5 accuracy. Also in [210], as well as performing food/non-food experiments, food group classification was performed. A CNN was developed that was trained using extracted segmented patches of food items. The food items used in this work were based on 7 major food types. The patches were then fed into a CNN using 4 convolutional layers with different variations of filter sizes and using 5 x 5 kernels to process the patches. Results in [210] achieved 73.70% accuracy using 6-fold cross-validation. These studies confirm that CNN provides an efficient method for food image recognition to provide for accurate food logging to promote obesity management. These results from published works show that using CNN enables for efficient food image classification and the remainder of this section discusses what food image classification areas CNN approaches have been applied for food image logging.

2.14.3 Deep Feature Extraction for Food Image Classification

Recent research has used deep features extracted from pretrained CNN architectures to train machine learning classifiers for food image classification. Figure 2.21 is a diagram describing the pipeline for deep feature extraction for automatic food image logging. Some research has opted for deep feature extraction opposing to fine-tuning pretrained CNN or training from scratch because less computational power and time is needed or datasets that are used are small. Well-known CNN architectures (e.g. AlexNet, VGG-16, GoogLeNet) have been used for deep feature extraction in classifying food images to automate food logging. Fig 2.22 and Fig 2.23 are examples of deep features extracted using GoogLeNet pretrained CNN. These features can then be used to train supervised machine learning algorithms

for image classification. The remainder of this section will discuss research that use deep feature extraction to detect food in images and classify food items in images for automatic food logging. A comparative review was carried out on analysing the performance of some pretrained CNN architectures [214]. This review used VGG-S, Network in Network (NIN), and AlexNet for deep feature extraction to train food detection models. Food/non-food image dataset was collated, and deep features were extracted from the models to train machine learning classifiers (one-class SVM classifier and binary classifier). Results showed that binary SVM classifiers trained with deep features achieved 84.95% for AlexNet, 92.47% for VGG-S, and Network In Network model achieving 90.82%. It is worth noting that UNICT-FD889 dataset used for deep feature extraction in [214] contains minimal noise as the images are focused on the food item. Therefore this may contribute to high accuracy results. Further work could be completed in utilising a larger food image dataset consisting of images from different environments and also using different machine learning classifiers for further comparison.

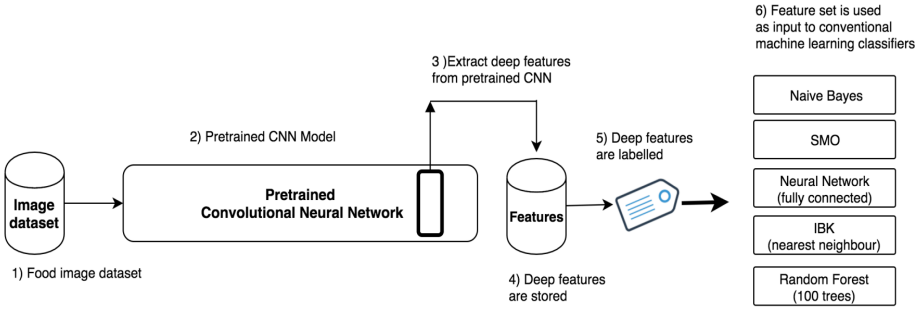


Fig. 2.21 Figure describing deep feature extraction process using pretrained CNN. Deep feature extracted from CNN can be used to train machine learning classifiers for image recognition. [26].

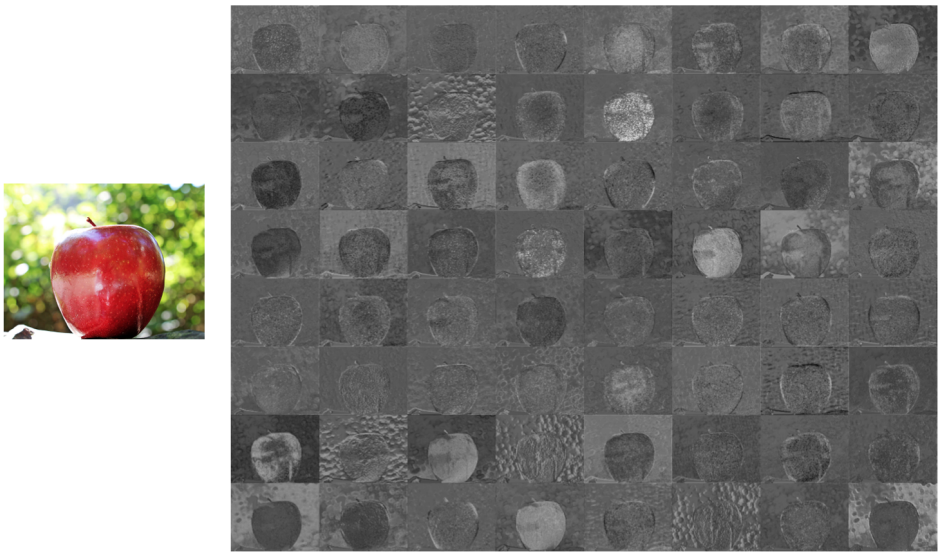


Fig. 2.22 Output of convolutional layer activations can be analysed using an input image. Each layer of a CNN consists of 2D arrays of channels and they are able to illustrate what areas or features of an image are 'activated'. Fig 2.22 illustrates the image features activated using GoogLeNet pretrained CNN model using layer 'inception_3a-1x1'.

.

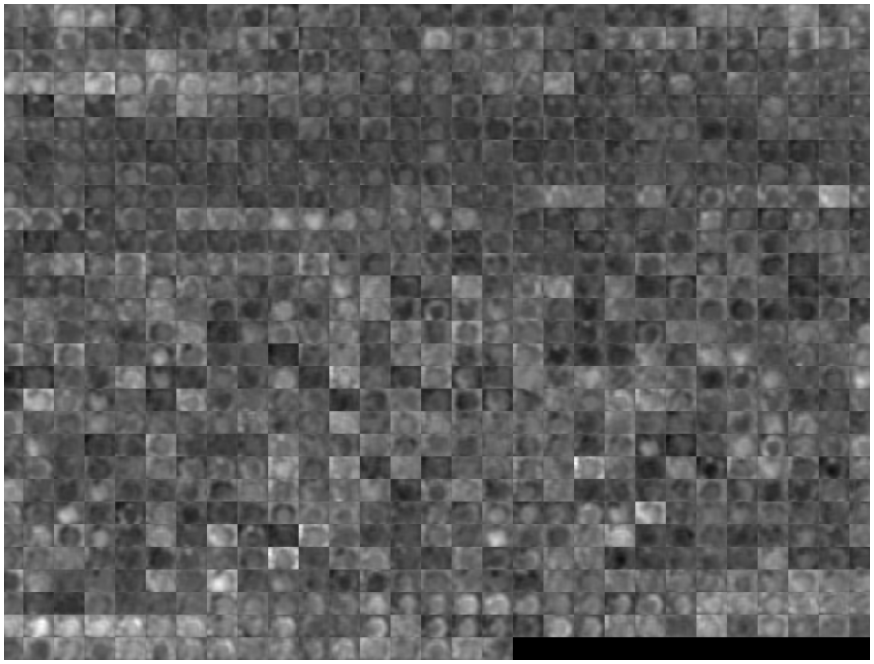


Fig. 2.23 Example of activations located deep in GoogLeNet convolutional layer 'loss3-classifier', the network learns to detect more complicated features. Deeper layers combine features from earlier layers to highlight detailed shape and features.

Other research also explored the effect of training machine learning classifiers using deep features extracted from different layers in using a pretrained AlexNet architecture [215]. Authors used AlexNet model to extract deep features from various layers deep in the CNN architecture (FC6, FC7, and FC8 layers). The food image dataset used in [15] was PFID. Two experiments were presented in [215]; classifying high-level food categories by organising PFID dataset into 7 category dataset and also classifying individual types in PFID (61 classes). Results showed that the highest accuracy for the 61 class dataset was 70.13% using deep features extracted from layer FC6 in AlexNet. For the 7 class dataset, the highest accuracy achieved for deep features was 94.01% using layer from FC6. The contribution in [215] supports the same findings in [214] suggesting that deep feature extraction provides high accuracies in classifying small grouped food image datasets (related food items) as well as datasets with specific different food types. Results also suggest that AlexNet deep features are able to efficiently generalise between high level food groups and also classify specific food groups with reasonable accuracy. However, more research needs to be completed in using deep features to classify food images in real world environments as PFID used in [215] was a laboratory prepared dataset. As AlexNet is an early CNN architecture with a small amount of layers in comparison to more recent models, it was able to achieve reasonable accuracy in food item classification. AlexNet deep features from FC7 layer were able to achieve 57.87% using a standard linear SVM classifier classifying UECFOOD-100 and 43.98% in classifying UECFOOD-256 [216]. Fine-tuning AlexNet on a food image dataset and then performing deep feature extraction improved the accuracy to 67.57% in classifying UECFOOD-256.

GoogLeNet Inception CNN has also been used for deep feature extraction for food image classification [217]. Authors fine-tune a pretrained GoogLeNet model using a food image dataset, and then deep feature extraction was used on another food image dataset. Experiments were completed in training a SVM using GoogLeNet deep features, in which

the GoogLeNet model was fine-tuned using a food image dataset. Results showed that using deep features with SVM with PCA trained using fine-tuned GoogLeNet features achieved 95.78% in classifying RagusaDB test set and 98.81% in classifying FCND test dataset which was an increase in accuracy compared to other works using same datasets. Using RagusaDB and FCD combined together for experiments achieved 917.41%. The datasets used in [217] were small and more comparative research is needed in using a larger dataset of images photographed in different environments and real-world settings to evaluate the proposed approach [217] fully. It is evident from the literature that deep feature extraction using CNN for food detection and food item recognition is able to support automated food image logging. Table 2.2 gives a summary of previous research that has utilised CNN for deep feature extraction to classify food images. It is clear from the research discussed that deep feature extraction and fine-tuning of CNNs had enjoyed great success in food image logging for dietary management.

Table 2.2 Selection of research that used Deep CNN feature extraction methods for food image classification.

Deep CNN Feature Extraction for Food Image Classification				
Author	CNN	Dataset Type	Food Dataset	Acc %
VGG-S [214]	VGG-S%	2 (Food/NonFood)	RagusaDB	92.47%
NIN [214]	NIN[214]	2 (Food/NonFood)	RagusaDB	90.82%
AlexNet [214]	AlexNet	2 (Food/NonFood)	RagusaDB	84.95%
GoogLeNet [220]	GoogLeNet	2 (Food/NonFood)	RagusaDS	94.67%
GoogLeNet [220]	GoogLeNet	2 (Food/NonFood)	FCD	99.01%
NIN [212]	NIN	2 (Food/NonFood)	IFD	95.1%
AlexNet [215]	AlexNet	7 (food groups)	PFID	94.01%
AlexNet [216]	AlexNet	100 food items	UECFOOD-100	78.77%
AlexNet [216]	AlexNet	256 food items	UECFOOD-256	67.57%
VGG-19 [218]	VGG-19	101 food items	RawFoot-DB	97.69%

2.14.4 Fine-Tuning CNNs for Food Image Classification

As discussed, fine-tuning involves taking a pretrained CNN model (e.g. AlexNet or GoogLeNet) and retraining the final layers to solve a new problem. Fine-tuning provides the opportunity of using state of the art CNN architecture, already configured with parameters, to provide for accurate image classification which surpasses the use of handcrafted features. There is also the advantage of timing when fine-tuning CNN models, because only the final few layers are being trained the time taken to train the model is dramatically reduced. This method of fine-tuning CNNs has been applied to food image classification with promising results, for example in [218] a GoogLeNet Inception V3 model was fine-tuned to classify a variety of difficult food image datasets (UECFOOD-100, UECFOOD-256, Food-101) and results show that using Inception V3 fine-tuned achieved a top-1 accuracy result of 88.28% for Food-101, 81.45% for UEC-100, and 76.17% for UECFOOD-256. Similar research also used a pretrained GoogLeNet Inception model and achieved a top-1 accuracy of 76.3% [167]. Previous examples already discussed utilise fine-tune models to predict if a food image contains food using a fine-tuned GoogLeNet model and achieving 99.2% accuracy in classifying images into food or non-food [209]. In [219], researchers fine-tune a VGG-16 model to classify 15 food items from UECFOOD-100 dataset, the contribution of this research is the ability to implement a calorie estimation method along with prediction using VGG-16 model. The calories for each of the 15 meal types were provided by using commercial online recipe websites. The multi-task CNN proposed by [219] classifies the image and outputs the calories simultaneously. The authors used stochastic gradient descent (SGD) with a learning rate of 0.001. Results show that top-1 accuracy for classification was 82.48% and 97.45% for top 5 accuracy. Research discussed that fine-tuning a pretrained CNN model for food image classification achieves state of the art performance using a variety of food image datasets.

2.14.5 Calorie Estimation Using Deep Learning Based Approaches

In [234] research was completed that presented a food image dataset containing 2978 images across 19 food types and also provides the volume, mass, density, and energy associated with each food image. Each image in the dataset contains images photographed from the side view and also a top view using a smart phone, and a coin was present in each image for calibration. Authors in [234] combine the use of a R-CNN model and GrabCut to locate and segment the food portion to determine volume and then nutritional content. Dataset was partitioned into training and testing, and results show that mean error between estimated and true volume did not exceed 20% except for banana, grapes, and mooncake. From the research discussed, it is clear there is a need for accurate food segmentation and nutritional information calculation for accurate food logging. Previous research has utilised computer vision based approaches to automatically segment food portions however some issues may occur that could affect the accuracy of food portion segmentation (food overlapping, hidden food portions). In this study, deep learning combined with GrabCut is presented, and results are promising in achieving automatic food logging using food images. Similar research also use multi-task based CNN approach for simultaneous estimation of food categories and calories using food images [283]. Results from [283] show that a multi-task CNN based approach outperforms a single-task CNN approach with 94.14 absolute (abs) error calories for multi-task CNN and 105.73 abs calorie error for single-task CNN [283].

Similarly authors in [282] also apply a Faster R-CNN to detect multiple food items and to also output calorie content. Another CNN model was used to estimate calories and this CNN model was trained using a calorie dataset based on previous research authors completed in [283]. R-CNN model is used to detect and crop food images for each food portion to be then sent to calorie CNN for calorie estimation. To test this system, 2 food image datasets were used,(1) Annotated School Lunch Dataset (SLD)and (2) UEC FOOD-100. Classification results were 90.7% Mean Average Precision (mAP) for annoted SLD and 57.7% mAP for

UEC-FOOD100 dataset. For calorie estimation using CNN model, SLD was used and results show a mAE of 136.8 calories. For future work authors seek to combine models to create a multi-task CNN for food classification and also calorie estimation. Other works combine augmented reality (AR) with deep learning for calorie estimation [281] , DeepCalorieCam was proposed. The application is able to detect calories from dishes in a video stream, YOLOv2 CNN architecture was used to detect the dish within a food image and detected food dishes are then cropped to then be sent to another CNN model to determine calories. Further testing is needed with the application proposed in [281], especially with images and videos captured in free-living environments. From research discussed, deep learning in particular CNNs show to have massive potential in determining calories in meal images in comparison to traditional image processing and computer vision approaches. Much of the work discussed utilise a multi-CNN approach for calorie detection; a CNN for food detection and classification and another dedicated CNN is used for calorie estimation. Other work utilise seperate layers within a single CNN for certain tasks. More work needs completed in utilising state-of-the-art CNN architectures such as residual based CNNs, for multi-tasked food classification and nutritional estimation.

2.15 Food Image Datasets for Dietary Management

There are a number of food image datasets that have been developed to explore how image processing and computer vision approaches can be used to predict meal items in images. This section will discuss popular food image datasets that have been used in previous works. Figure 2.24 are example images from each food image dataset.

2.15.1 Food-5K (Food Detection)

Food-5K dataset consisted of 2 categories; food and non-food, training is balanced and contains 1500 images of each category [209]. The dataset also contains a validation and evaluation set and each category contains 500 images each per dataset. The authors developed this dataset to measure the performance of using GoogLeNet pretained CNN for classification. Food-5K was developed by selecting images from already public available datasets e.g. Food-101 [213], UECFOOD-100 [257] and UECFOOD-256 [258]. The authors described this dataset as being varied as they selected foods that cover a wide variety of different food dishes. The images also contain some noise and multiple food items may be contained in an image. The non-food images consisted of images that do not contain food items (objects or humans). Food-5K was used to find out how ResNet-152 deep features perform in detecting food items in images, which can be argued is an important first step in food image classifcation for food logging. The authors developed the non-food image dataset from using other publicly available datasets e.g. Caltech101, Caltech256, Emotion6, and Images of Groups of People.

2.15.2 Food-11 (Food Types)

Food-11 is a dataset that comprises of 11 major food groups [209]. The 11 categories are diary, bread, egg, dessert, meat, fried food, pasta, seafood, rice, vegetables/fruit, and soup. Food-11 dataset was also created using images from Food-101, UECDOOD-100, and UECFOOD-256. The authors of Food-11 stated that the images selected cover a wide range of food types in order to train a strong classifier that had the ability to classify different varieties of foods. Many of the images contained in Food-11 were taken in real world environments, therefore the images contain high colour variation and some noise (unrelated food items) may be present. The developers of this dataset have divided the dataset into

training, validation, and evaluation similar to Food-5K. Food-11 was used to explore the performance of ResNet-152 deep features in categorising food images using Food-11.

2.15.3 RawFooT-DB (Food Texture)

RawFooT-DB [259] food image dataset was developed to research the use of computer vision methods to classify food image textures under different lighting conditions. Each image in RawFooT-DB is unique in regards to the light direction, light intensity, and colour illumination and food image textures are isolated with no noise or other food items present. The dataset contains 68 classes with wide variety of food types ranging from fish, meat, fruit, and cereals. RawFooT-DB dataset contains tiles from the images in the RawFooT-DB. Each image is divided into 16 tiles, 8 tiles are for training and the remaining 8 for testing. Each class contains 368 images (tiles) which represent 8 tile texture samples under 46 different lighting conditions. In this research, we explored the use of ResNet deep feature features to train machine learning classifiers. RawFooT-DB was used to explore how ResNet-152 deep features perform in generalising food texture between class variance. Previous research divided RawFooT-DB into different lighting condition subsets [259], in this work we explored the performance of using ResNet-152 deep features across multiple lighting conditions and each food class in RawFooT-DB contains multiple food texture patches across different lighting conditions.

2.15.4 Food-101 (Specific Food Items)

Food-101 consists of 101 food categories and each category contains 1000 images [213]. The Food-101 dataset have been described as challenging as much of the images in the dataset contain noise and the images were collated from Foodspotting, which is a social media website that allows users to upload food images. This means that images used are from a real-world setting i.e. restaurant or at home and not in a lab environment. Food-101

allows us to research how ResNet-152 deep features performs in classifying food items with similar food dishes in varying real world environments. Authors of Food-101 specify dedicated training and testing splits with testing splits containing images that are 'cleaned' of noise, in this work we also use 75:25 training/testing partitions, however data was shuffled before partition for preliminary analysis to determine how ResNet-152 features perform in classifying images with noise and intense colour and food variation. Figure 5.2 illustrates an example of the images in the datasets.

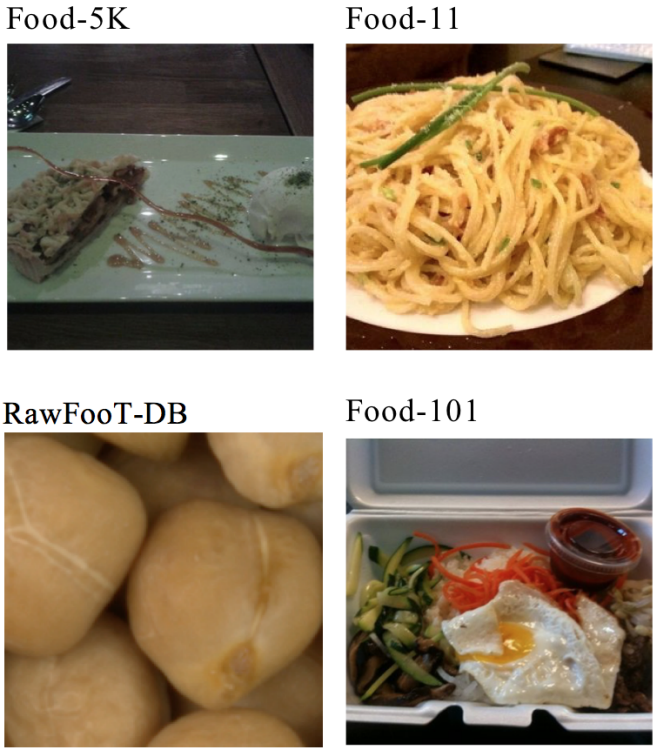


Fig. 2.24 Food image datasets used in previous published research for food image classification.

2.16 Summary of Literature Review

- Research suggest that using digital methods for food logging such as smartphone applications or websites can improve retention and usability.

- Using images for dietary management is an advantageous approach to food monitoring as more information regarding portion size and is much easier to document when compared to traditional approaches.

- Computer vision methods can be used to automate food logging for dietary management through predicting food items in images and also to measure portion size for nutritional estimation.

- Computer vision approaches can also be used to automatically segment food portions to isolate food portions to enhance food classification.

- Deep learning methods such as CNNs are gaining increasing popularity due successfully be used classify and segment food images accurately when compared to traditional approaches.

- Deep feature extraction using pretrained CNNs can be combined with conventional machine learning classifiers to predict food items in images and achieve high accuracies when compared to traditional feature extraction methods (SURF, HOG).

- Crowdsourcing can be used to rate determine nutritional content in food images for dietary management.

- Based on the literature, there is potential to combine automation within RFPM by incorporating food identification using machine learning approaches such as deep learning. RFPM method could be combined with deep learning and crowdsourcing to create a dietary management pipeline. Deep learning and computer vision approaches would be able to accuately determine food type and volume, and a crowd of users could be used to determine nutritional content.

2.17 Potential Contribution

Using the literature, the following potential contributions areas have been highlighted in regards to determining calorie content in image food portions. More research needs completed in utilising image statistical information with in a food portion. Exploratory work could include experiments that correlate pixel information with calories to determine calories in food images. As well as this, crowdsourcing has been shown to be useful in determining healthiness of food items and meals, which could be harnessed to support food logging, however more research is needed to exploring how crowdsourcing calorie estimations from users can be used in calculating nutritional content of food items in photographs. In regards to food image classification, computer vision methods have been successfully applied, however more experiments is needed in determining what feature extraction combination approach is best suited in classifying food images captured in free-living environments. Further research will focus on developing and evaluating computer vision feature fusion approaches for food detection and to predict food image types in real-world environments using a variety of machine learning classifiers. Furthermore, state-of-the-art CNNs will be investigated and evaluated in regards to the generalisation capabilities of deep feature extraction in detecting food in images and predicting food types in images as well as assessing what machine learning classifiers are best suited in classifying food images using deep CNN features. Another potential contribution would be to compare the performance of fine-tuned state of the art pretrained CNNs with deep feature extraction for food detection (e.g. ResNet-152), food group classification, and specific food item classification. A further potential contribution would be to combine these elements together by proposing a dietary management system that uses crowdsourcing and computer vision approaches for calorie estimation and food image classification.

2.18 Conclusion

Food image logging provides a opportunity to determine more information from food items to enhance the accuracy and convenience of food logging in comparison to traditional food logging approaches. This chapter discusses the technologies that have been used with the aim of automating food logging using artificial intelligence (AI) and supervised machine learning approaches. Previous work have utilised the use of global and local features for food image classification to use with machine learning classification algorithms and more recent methods such as CNN flavour of deep learning for image classification . This chapter discusses feature extraction algorithms that have been used in computer vision for food image logging. It is clear from the literature that using a segmentation based or semi-automated approach to isolate food portions is able to increase classification accuracy. In regards to deep CNN features for food classification. Research is needed in using new CNN architectures for deep feature extraction such as the ResNet architecture for detecting food images, food groups, and also specific food categories. Using CNN deep features to train machine learning models along with semi-automation food segmentation needs to be explored and benchmarked against previous methods discussed in literature. The literature also suggests that crowdsourcing can be used to provide dietary support. More research is needed in determining how crowdsourcing can be used to provide accurate nutritional intake estimates for food logging and determine statistical relationships between calorie estimation for different meal types. There is a need to combine multiple approaches such as deep learning for food classification and crowdsourcing for nutritional content in food items to support dietary management. The literature review also highlighted potential areas for investigation in regards to food image detection and classification as well as calorie estimation.

Chapter 3

Semi-Automated Estimation of Calories of Meals in Photographs

3.1 Introduction

Accurate food logging is an essential method for dietary management. One of the main reasons for weight gain can be attributed to an increase of food portion sizes [223, 224]. As discussed, traditional methods for calorie estimation has been to either document nutritional content using food packaging or search online calorie tables/ databases. However, users would have to know the exact portion size of the food item to determine accurate nutritional content. The underestimation and overestimation of consumed food items is an area of dietary management that needs addressed, and the use of food images allows for more information to be ascertained to determine portion size. In regards to calorie estimation and food segmentation methods, much research has been completed that utilise manually segmented methods, automatic and a hybrid approach. As well as manual image segmentation techniques, automatically segmentation approaches have been extensively applied to food images for dietary management. These approaches are discussed in detail in Chapter 2

Literature section 2.12. It is clear from the literature review that more work is needed in using local image data as a contributing factor in calorie estimation. The work presented in this Chapter presents a method using image pixels as a means to generate calorie content information. The remaining content of this Chapter is as follows: Section 3.2 states the aim and objectives of this study. Section 3.3 discusses related work in regards to image calorie analysis and food diary log interventions that have already been implemented. In Section 3.4 the study methodology is presented that describes the data collection procedure to build a ground truth dataset. In Section 3.5 preliminary results are presented of the experiments. Section 3.6 is a discussion of the main research and clinical implications of this research and 3.7 is a discussion of the implications of this study in regards to dietary management. The work presented in this Chapter is based on published works [225].

3.2 Aim & Objectives

- The aim of this research study was to investigate image segmentation approaches for food calorie estimation.

  The objectives of this study were:

  - To establish a ground truth dataset of food portions.
  - To correlate image pixels with weighted food portions using image processing approaches.
  - To develop a statistical model to predict calories within food portions.
  - To evaluate statistical model on test food portions.

3.3 Methodology

3.3.1 Data Collection

To achieve the aim of this Chapter a ground truth dataset was collected. A food item was selected (uncooked sweetcorn), and different portions were measured out in grams using a food scale. The portion of food would was placed on a plate and beside the plate was a fiducial marker (1cm^2 square) [20]. The number of calories per portion of the food item was noted using the food-packaging label. A photo was then taken of the food portion along with the fiducial marker next to the food portion, and the purpose of the fiducial marker was to provide context awareness that will help measure the area of the food portion. This process was repeated five times for different portions of the same weight, and a photograph was taken of every portion (five portions of 10 grams = five photographs). The 1cm^2 fiducial marker was placed directly next to the food to attain a more accurate measurement for ground truth data. However, in a free-living environment, the fiducial marker would be placed next to the plate as opposed to being directly beside the portion. An iPhone 6 smartphone was used to take a birds-eye image of each food portion. Food portions photos were not captured in a controlled environment. An interface was developed that allowed users to segment the food portion using a polygonal tool and the application also detected the one-centimetre square using Hue Saturation Value (HSV) colour thresholds to measure food portion area. This process is described in Figure 3.1.

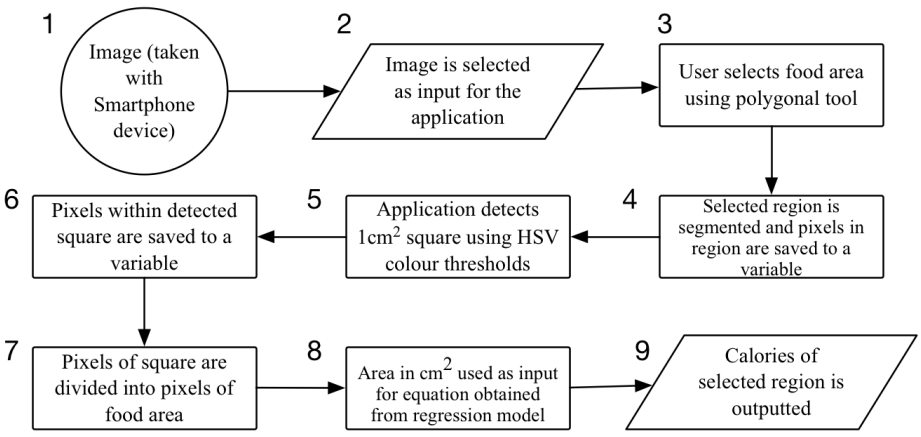


Fig. 3.1 Flow diagram describing main processes in pipeline to segment image using colour thresholds and calculate area.

3.3.2 Food Area Calculation

The food area calculation method was able to output the area of the food portion by calculating the number of pixels in the food portion and dividing it by the number of pixels in the 1cm^2 fiducial marker. As Figure 3.1 states, users can select an image. Users then select a region of interest using a polygonal tool. This region is segmented and saved. The application was able to detect the 1cm2 square using HSV colour thresholds. This process was developed by firstly taking photos of the fiducial marker in different lightings. The colour in the square for each picture was highlighted using a colour threshold segmentation function. Thresholds were determined using the colours highlighted in the images captured under different lighting conditions. Once the fiducial marker was detected, it is then segmented and pixel amount was stored. The area was computed by dividing the pixels in the food portion by the number of pixels in the fiducial marker.

$$\bar{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} \frac{\sum PixelinFood_i}{\sum PixelinReference_i} \tag{3.1}$$

This process is described in equation 3.1 where $PixelinReference_i$ refers to the amount of pixels within the detected 1cm^2 fiducial marker. $PixelinFood_i$ refers to the pixels within the food region segmented by the user. The pixels in the food region are divided by the pixels in fiducial marker to attain area. This is repeated for n instances. The mean is then computed using each area calculated to ascertain an average area for a portion size.

3.3.3 Linear Regression Calorie Calculation

After the dataset was collected for the food item, a regression model was then developed. A regression model was used because it can be used to illustrate relationships and trends among different numerical datasets. For this work, the area of food and calories was plotted on a linear regression model to highlight the relationship between portion size and calories. Once the regression model was developed using the ground truth portion calorie dataset, a defining formula was extracted from the regression model. The linear regression model formula describes the relationship between the two variables as shown in equation 3.2. This formula was used to predict calorie content of a given area value. This linear regression formula was included within the developed system to test new images of the food item to determine calorie content.

$$y = a + bx \tag{3.2}$$

To test the methods outlined, an experimental approach was used, and sweet corn was chosen as a food item. Ten-gram increment portions were weighed, and 5 photos were taken of each along with the 1cm^2 fiducial marker next to the plate. This process was repeated five times per portion (five photographs of each ten-gram portion, etc.). The developed application was then able to allow the user to use a polygonal tool to draw around the food

portion. The application would segment the food portion and save the image. The application would also take note of the pixel count within the segmented region. The application would then also detect the 1cm^2 fiducial marker using HSV colour thresholds as the fiducial marker is a specific colour. Figure 3.3 is an example of both regions segmented.

3.3.4 Evaluation Methods

To evaluate the system outlined in this chapter, a number of test portions were measured. Test portions were measured the same way in which the training dataset was developed and each test portion was photographed along with the same context reference point used with the training dataset. The regression model developed using the training dataset was used to predict the calories in the test portions using a semi-automated approach.

Fig. 3.2 Image of food portion with $1cm^2$ fiducial marker next to plate.

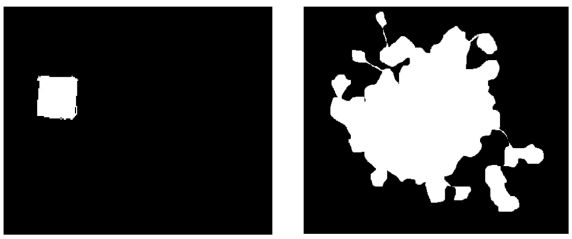


Fig. 3.3 Image of segmented regions; $1cm^2$ and food portion.

3.4 Results

For the training dataset, the pixels in the food portion were then divided by the pixels in the detected 1cm^2 fiducial marker to reveal portion area. An average area was calculated for each portion by using the area pinpointed in each of the five photographs taken of each portion. The data was then plotted on a regression model to illustrate the relationship to reveal a positive correlation. As stated, the regression model is able to describe the relationships between the two datasets. Regression model would allow us to predict the calories of a given area by giving X a value which is the area of the food portion. Equation (3.3) is the equation defined using the regression model.

$$y = 0.7054x - 6.16 \tag{3.3}$$

3.4.1 Test Portion Results

This equation was then included within the proposed system. Once the system defines the area of the portion the user selected, this then could be used as input for the equation 3.3 as X to output the calorie content of that selected region. A series of tests were completed using a new batch of images captured using sweet corn portions. Each portion in each new images was measured at 10-gram increments starting at 15 grams. Table 3.1 lists the calorie content using the application and the actual calorie content using the food item packaging. Table 3.1 illustrates that the system described in this study is able to determine calorie content of a kernel based sweet corn portion.

Table 3.1 Results of area of food portions and calories using model.

Weight Portion	Predicted Calorie	True Calorie Content	Percentage Error
15g	11.78	12	1.83%
25g	18.41	20	7.95%
35g	24.33	28	13.90%
45g	29.43	36	18.5%
55g	36.2	44	17.72%

3.5 Discussion

The work described in this Chapter discusses the use of semi-automation to accurately predict calorie content in photographs. The work presented in this chapter was inspired by the need to remove much of the complexity of food image classification and calorie prediction by incorporating a polygonal segmentation tool. Research has been completed in segmenting food portions using computer vision approaches, with varying results and much segmentation works have focused on using laboratory prepared images. In this chapter, we decided to explore how semi-automation, using a polygonal tool can be used to accurately segment a region of interest (ROI) and calculate area based on pixel quantity and reference point. The idea of utilising a reference point has also been explored in previous works [1] to determine portion size, however, in this work, a reference point is used to calculate the area in the food portion by dividing the food portion pixels by the pixels in reference point and to reveal area cm^2.

This application was tested using different portions of the same food item, and each test portion area size was not used in the training dataset to build the regression model. The ground truth calories for each weighted portion were determined using food item packaging. The same process described earlier was then completed for each image to determine calorie content. The reason why the weight was taken into consideration was that the calorie content

could be determined using the packaging contents stating calories per 100 grams. This could be used to compare the experimental results and ground truth data. Results show that the application can ascertain the calorie content with an average percentage error of 11.82%. The error percentage for each portion accounted for only a small number of calories as shown in Table 3.1 due to the small number of calories already existing in the food item. The depth of each portion could also be taken into consideration in future work using technologies such as 3D depth cameras as used in related research [233]. This would give a more accurate reading when used during the data collection section.

The purpose of the work presented in this chapter was to address accuracy issues of users self-reporting the nutritional content of meal items [284]. In many food logging applications, the user must recall the food item as well as the portion size (cup, tablespoon, gram). Many users may not know the exact portion size is on their plate [284], which may lead to inaccuracies and wrong portion size being recorded. The developed system allows users to use a tool to manually draw around a food portion to calculate the area (ensuring that the reference point is placed next to the plate to give context awareness to determine portion size). The system then outputs an approximation of the portion's calorie content. This process outlined in this Chapter is more convenient for the user and user-friendly because if the user is using an online API to determine calorie content, the user may not visually know what the correct portion size the food item in the photograph may be leading to inaccuracies. The use of a polygonal tool to manually segment the food item can remove much of the computation power needed and can also improve accuracy by allowing the user to determine exact food items.

3.6 Key Findings

This chapter proposed a novel approach that allowed users to manually segment food portions to calculate calories. A regression model was trained using a ground truth dataset that was collected through correlating image pixels with weighted food portions. Food portion sizes (area cm2) were estimated using a colour segmentation approach by dividing the number of pixels in food portion by the number of pixels in a reference object. The statistical regression model was evaluated on test food portion images, and results of experiments show the regression model can accurately calculate calories with a mean percentage error of 11.82%.

3.7 Implications for Dietary Management

The methods presented in this research study described how a semi-automated approach could be used to personalise food logging. Traditional methods for food logging allows the user to search nutritional information on a food item using an online database. Many smartphone applications also use online API to search for calorie content of popular food brands. However, this may lead to inaccuracies as the user may not know the exact portion size of the food item. The methods presented in this study allow the user to draw around a food portion to compute nutritional content for more accurate calorie calculation for food logging. As well as this, if the user knows in advance the entire food portion will not be consumed, then a semi-automated approach using a polygonal tool would enable the user to specifically highlight food portion that will be eaten for exact nutritional intake calculation. This work also alleviates the burden of overestimating and underestimating portion sizes and promotes usability as the system can automatically calculate nutritional information based on the food area.

3.8 Summary

The research discussed in this Chapter focuses on using local statistical image information to calculate calories in a specific food portion. This work also highlights the importance of logging accurate food intake using images, which is useful in providing accurate food logging for dietary management, in the prevention and management of obesity. The methods outlined in this study present a system that allows users to measure food portions to pinpoint calorie content. This is vital in allowing individuals to manage calorie intake to promote healthy living. Other research use automatic segmentation methods but this have been shown to be difficult due to the nature of different food types. A manual segmentation approach would allow the user to draw around a food portion for greater context and portion estimation accuracy. Other image analysis applications researched use remote food photography method, which involves the user taking an image of food, to then send the image to another location for nutritional analysis. However, this is may be inconvenient as it takes a certain amount of time to complete the process as a dietician or nutritional expert will have to analyse the image to for the nutritional content [62] therefore an automated approach for calorie calculation is needed. This is similar to crowdsourcing initiatives discussed in [40] as time is still an issue. Other research use an index based scoring system based on dietary guidelines that can segment a user's meal up into different defined food types such as dairy, vegetables, and protein. The user can add meal portions according to key meal components for quick entries [29]. Other research utilised the use of NFC scanners to scan RFID tags on food packaging to attain nutritional content [25]. However, there is still a need for higher accuracy in regards to calories consumed by the user. This system presented in this Chapter describes and demonstrates a process in which accuracy can be achieved for particular food items. The correlation of food image pixels with calorie content to determine portion size has potential. For ground truth datasets, it is important to gather calorie readings and portion sizes to build

multiple regression models for different food items. This work presents a novel approach tested on a single food item, however, more work needs to be completed in applying these methods to different food types to test further feasibility.

Chapter 4

Automated Adjustment of Crowdsourced Calorie Estimations for Accurate Food Image Logging

4.1 Introduction

Recent research has utilised crowdsourcing to provide for dietary management [185, 186]. One of the main challenges for dietary management is accurate calorie estimation. Current food logging methods consist of either using paper diaries or smartphone/tablet weight management applications; however, these approaches may be cumbersome, time-consuming, and calorie recording may be inaccurate in regards to documenting portion sizes [49]. Research has utilised computer vision and image processing methods to determine calorie content and portion size in images to circumvent the shortcomings of current food logging approaches with promising results. Research has utilised calibration reference points situated next to food portions to determine portion size for accurate calorie calculation, and crowdsourcing has also been utilised to provide for accurate portion size and calorie

estimation for dietary management [166, 167, 168]. Previous research has incorporated crowdsourcing techniques to allow multiple users to determine portion size and food type and these responses are aggregated to determine calorie estimation and other research has used crowdsourcing to rate healthiness of food images to aggregate opinions and other methods which have been discussed in Chapter 2 Literature Review. Crowdsourcing and remote food photography method (RFPM) have also been combined to propose an automated food identifying and calorie estimation pipeline also with promising results [40]. The rise of camera-enabled smartphones and portable technology usage has enabled users to capture food portions for food logging to ascertain greater detail regarding portion size and composition and also to share on social media websites [51]. There is great potential in using computer vision technologies with crowdsourcing techniques to provide for accurate food logging. However, more work needs to be completed in utilising crowdsourced estimates in determining nutritional intake of meal images. This work investigates the use of crowdsourcing calorie estimations from a group of experts (nutritionist) and non-experts in estimating calories in meal images and further investigates the use of crowdsourcing descriptive statistics to adjust calorie estimations for higher accuracy. The work presented in this chapter is based on published works [247].

4.2 Aim & Objectives

- The aim of this study was to investigate the feasibility of crowdsourcing non-experts and experts in accurately determining calorie content in images of meals from an existing calorie estimation dataset.

  The objectives of this study were:

  - To determine accuracy of non-experts in estimating calories in food images.
  - To determine accuracy of experts in estimation calories in food images.

- To apply statistical analysis to compare performance between experts and non-experts.
- To propose a crowdsourcing calorie adjustment algorithm to enhance calorie estimation accuracy.

4.3 Methodology

This chapter consists of two sets of experiments (1) compute descriptive statistical analysis (2) calorie adjustment process. The first set of experiments were completed to analyse the performance of non-experts and experts responses collected from a calorie estimation survey and determine the relationship between each group and meal images. Secondary experiments were completed that used the statistical metrics generated in the first experiments to adjust calorie estimations to enhance calorie accuracy. Accuracy results are then compared with ground truth calories of each meal image to measure the performance of the adjustment process. The experiments presented in this study are exploratory in investigating how food images and crowdsourcing can be used to provide dietary management support. For the calorie estimation survey, ethical approval was obtained by School of Communication Filter Committee at Ulster University.

4.3.1 Participants & Recruitment

Participants were invited to complete a calorie estimation survey. Participants were divided into two groups; experts and non-experts, experts were individuals who have knowledge of dietetics and nutrition, and non-experts individuals who have no trained knowledge of nutrition. Non-experts consisted of students within Ulster University and individuals not affiliated with Ulster University. Experts were recruited from the nutrition and dietetic staff from Ulster University. Convenience sampling was used (experts n=22 and non-experts

n=120). Survey responses that were partially completed or participants that measured their food items in kilojoules (KJ) instead of calories were not included in this analysis. An email was sent to students at Ulster University and detailed the purpose of the experiment and instructions. Participants were also asked to give consent before commencing the online survey. The online survey was approved by the Communication Filter Committee, Ulster University.

4.3.2 Online Survey & Food Images

The online survey consisted of 15 photographs of meals captured by a researcher who is also a trained dietician. The 15 meals included 5 breakfasts, 5 lunches, and 5 dinners. The photographs were captured on a smartphone device (iPhone 5). To calculate the calories of the food items in each image, each meal was weighed, and food labels and food tables were used. Participants completing the survey were asked the following question for each meal image From viewing the photograph, enter the number of calories you consider is in this meal? Kcal OR KJ. To complete the survey, participants were asked to input their estimated calories for each meal image as well as confidence levels. In this work, calorie estimations were only used for analysis.

4.3.3 Preliminary Descriptive Statistical Analysis

Descriptive statistics were generated using calorie estimations for each group e.g. mean, mode, and median. Standard Deviation for each meal image calorie mean for each group was computed for each group to further highlight differences. Calorie differences were also calculated for each of the participant's calorie estimation using the ground truth calories. Statistical analysis in this work was completed using Microsoft Excel version: 15.33. An analysis was also completed that compared crowdsourced descriptive statistics for each meal (mean, mode, median) to ground truth calories to determine what metric is most accurate.

4.3.3.1 Crowdsourced Calorie Estimations vs Individual Calorie Estimations

An analysis was also completed to determine if crowdsourced estimation metrics outperform individual participants estimations (non-experts). Non-experts were used in this analysis as there are more participants (n=120) compared to experts (n=22), this would allow for greater analysis and insight into using crowdsourcing to determine calories in food images. Calorie difference was chosen to measure performance between crowdsourced estimations and individual estimations. Calorie difference was calculated using participant calorie estimations for each meal image, and an overall mean calorie difference was computed (using ground truth calorie). For crowdsourced estimation metrics, the mean, mode, and median was calculated for each meal image type using participant estimations. Calorie difference for each meal type image (using mean, mode, and median metric) was also computed using ground truth calories. Finally, an overall calorie difference was calculated using ground truth and mean, mode, median metric for each meal image (which were calculated using participant estimations). Mean calorie difference for each participant was compared against the overall mean calorie differences calculated for mean, mode, and median for each meal image to determine if crowdsourced calorie estimations perform better than individual participants.

4.3.4 Calorie Adjustment Statistics

Secondary exploratory experiments were completed that used descriptive statistics calculated in first experiments to adjust calorie estimations to enhance accuracy. The purpose of these experiments were to determine if crowdsourced calorie errors could be used to adjust future calorie estimations to calorie enhancement. Calorie overestimations were crowdsourced from each participant estimation and used to deduct calories. To adjust the calorie estimates, the non-expert calorie dataset was used and a number of calorie statistics were first generated to adjust calorie estimates. The mean calorie estimate for each of the 15 meal images was generated. The overall calorie difference was also generated using all

calorie estimates. The calorie difference is computed by subtracting the ground truth calorie from the calorie estimate, this will reveal a calorie difference or error. This was completed for each calorie estimate in the non-expert dataset. The following equations describe how each metric was computed.

$$Mean\ calorie\ for\ each\ meal\ image\ Y = \frac{\sum_{i=1}^{n} E_i}{n} \tag{4.1}$$

$$Calorie\ difference = D = C_{est} - C_{gt} \tag{4.2}$$

$$Mean\ Calorie\ difference = \bar{x} = \frac{\sum_{i=1}^{D} E_i}{n} \tag{4.3}$$

Equation (4.1) describes how the mean calorie is computed for each meal image type in each training fold, where E_i is a calorie estimate and represents the mean calorie estimate for each meal image type and n is the number of estimations. Equation (4.2) describes how the calorie difference is calculated, where C_{est} represents the calorie estimation and C_{gt} is the calorie ground truth. Equation (4.3) is used to calculate the overall mean calorie difference using calorie estimations. Once has been calculated for each estimate using equation (4.2), $\bar{x}$is calculated which represents the mean calorie difference. Algorithm 1 states the pseudocode for the calorie adjustment process using crowdsourcing descriptive statistics, $\bar{x}$ and y . Figure 4.1 is a flow diagram that also describes the overall process of calorie adjustment.

Result: Calorie Estimate Adjustment

Mean calorie difference for each meal image = $Y = \frac{\sum_{i=1}^{n} E_i}{n}$;

Calorie difference = $D = C_{est} - C_{gt}$;

Mean calorie difference = $\bar{x} = \frac{\sum_{i=1}^{D} E_i}{n}$;

Test calorie estimate = u;

if $u > y$ **then**

  $a = u - \bar{x}$;

else

  u;

end

Algorithm 1: Adjusting calorie estimations using crowdsourcing metrics.

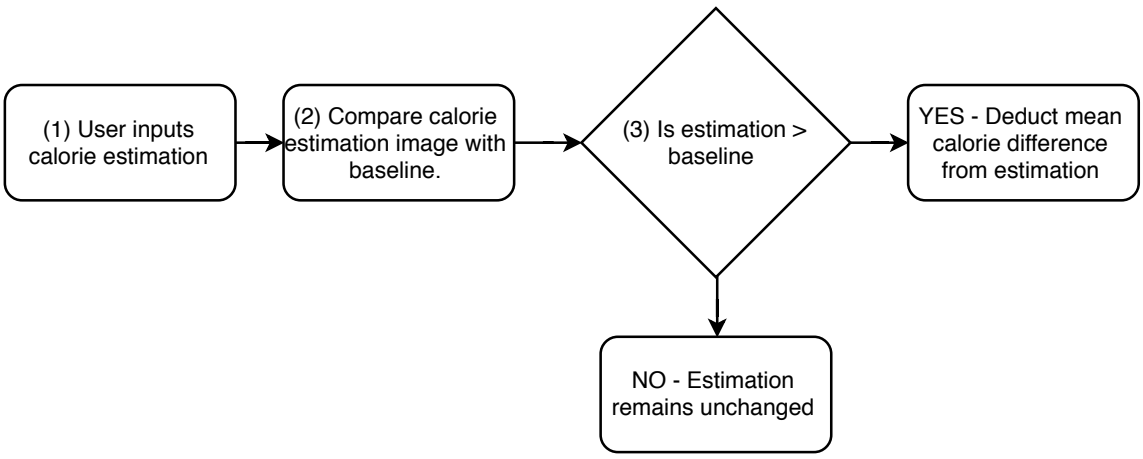


Fig. 4.1 Flow chart describing the process of calorie correction for dietary management using crowdsourcing.

4.3.5 Calorie Adjustment Evaluation

For calorie adjustment evaluation, 5-fold cross-validation was used to fully evaluate the proposed rule based system using the non-expert calorie estimation dataset. The non-expert dataset was used instead of the expert dataset as it is a larger dataset, which allowed us to evaluate the calorie adjustment process using 5-fold cross-validation. For 5-fold cross-validation, one fifth is using for testing and the remaining calorie estimations for generating statistics using calorie estimations. Test fold calorie estimations were adjusted using statistics computed using calorie estimations from training folds e.g. 24 participant estimations (one fifth) used for testing and remaining 96 participant calorie estimations for generating mean calorie estimations for each meal image type and an overall calorie difference. This process is repeated until each fold has been used as a testing split and the remaining for generating calorie statistics for adjustment. To evaluate the results of the calorie adjustment method, error percentages (equation 4.4) were calculated using the original mean estimation and ground truth as well the adjusted mean calorie and the ground truth for each meal image.

$$Error\ Percentage = \frac{C_{est} - C_{gt}}{C_{gt}}\ x100 \tag{4.4}$$

where C_{gt} is the ground truth calorie and C_{est}represents the original mean calorie estimation for each meal, the percentage error for the adjusted meal calorie estimation was also calculated.

4.4 Experimental Results

4.4.1 Descriptive Statistical Results

Descriptive statistics were generated using experts and non-expert survey calorie estimations for meal images. The mean, mode, median, and standard deviation were computed to describe the performance of each group in comparison to the ground truth. Figure 4.2 depicts the mean calorie estimations for each meal image for non-experts and experts group. The majority of mean calorie estimates for each meal image were greater than the ground truth for non-expert group, however, there is a strong correlation for true calorie content and mean estimation for the majority of meal images for non-experts with a Pearson correlation of 0.88. This suggests that a crowd of non-experts are able to determine meals that have a higher calorie content than others.

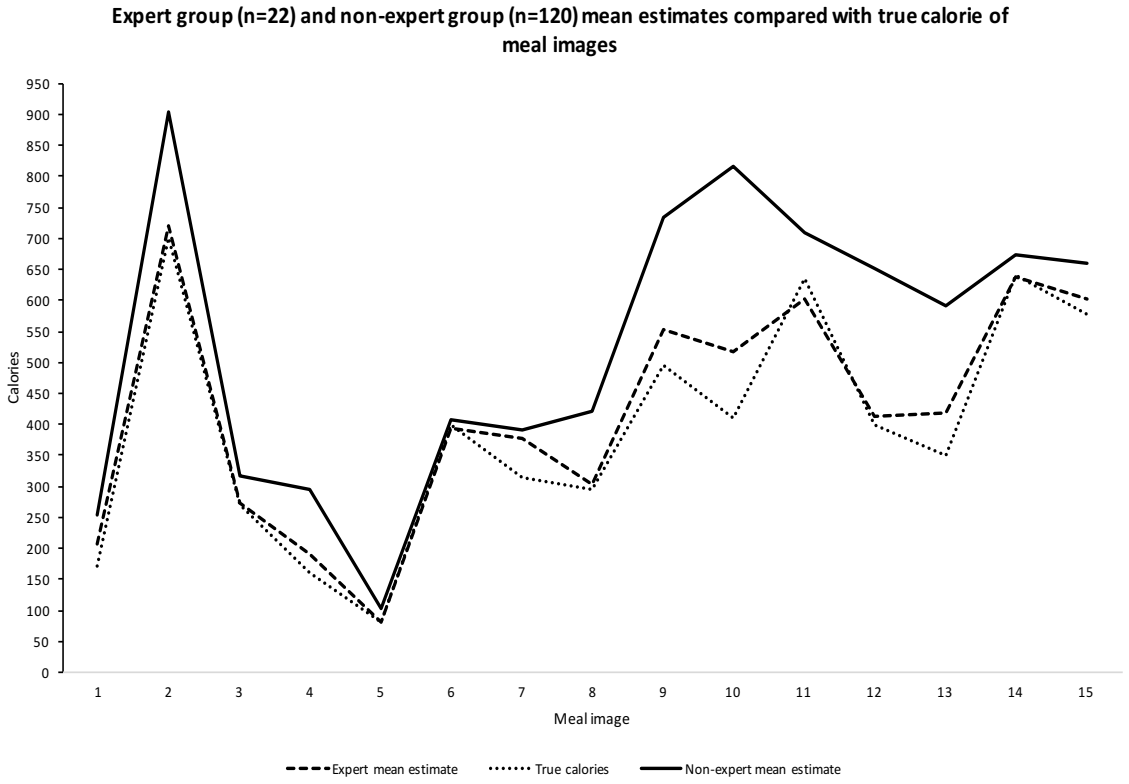


Fig. 4.2 Mean calorie estimation calculated for each meal image for each group compared with ground truth.

Figure 4.3 compares the mean standard deviation for the non-expert group for each meal image with the ground truth calories. Analysis using non-expert participants suggest that the higher the ground truth calories, the higher the standard deviation. Pearson coefficient was calculated to describe this relationship using the mean calorie estimation for each meal image and the standard deviation for each meal and results show that the Pearson correlation coefficient was 0.80, which also indicates statistical significance.

Table 4.1 is a list of the food meals and calorie amounts. Table 4.2 lists descriptive statistics for both expert and non-expert groups for each meal image. T-tests were carried out to determine if there was statistical significance between the non-expert calorie estimations for each meal image and expert calorie estimations for each meal image, these results are listed in Table 4.2.

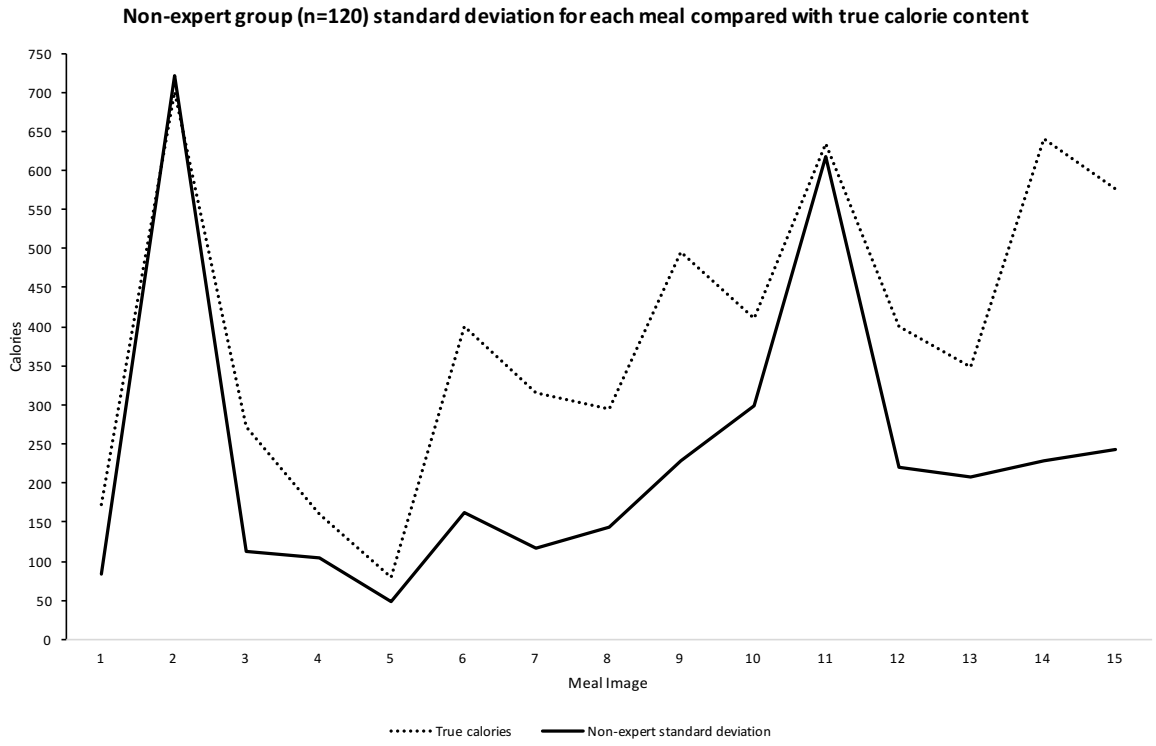


Fig. 4.3 Comparison of ground truth calorie for each meal and calorie standard deviation calculated using mean calorie estimation for each meal for non-expert group

Table 4.1 Meal types with calorie content used in online survey.

Meal Number	Meal Type	Calories (Kcals)
Meal 1	Bowl of cornflakes & semi-skimmed milk	172
Meal 2	Breakfast fry-up	700
Meal 3	Branflakes with semi-skimmed milk & banana	271
Meal 4	2 slices of white bread toast with butter & marmalade	160
Meal 5	Hard boiled egg	80
Meal 6	Bowl of stew	400
Meal 7	Egg with mayonnaise & watercress sandwich 2 slices wholemeal bread	315
Meal 8	Chicken, peppers, pesto basil leaves on white flat bread	295
Meal 9	McDonald's Big Mac	495
Meal 10	Fish and chips	410
Meal 11	salmon, champ, carrots, peas & sweetcorn & white sauce	634
Meal 12	Chicken and pasta with sauce topped with cheese	400
Meal 13	Pork chop, champ and broccoli	349
Meal 14	Spaghetti bolognese	641
Meal 15	Chicken curry with white rice	577

Table 4.2 Statistical metrics describing experts and non-expert estimations.

Meal Image	Non-expert mean (Kcals)	Non-expert mode (Kcals)	Non-expert median (Kcals)	Expert mean (Kcals)	Expert mode (Kcals)	Expert median (Kcals)	P-value (using expert & non-expert estimations)
1	252.88 (±83.48)	200	250	205.91 (±48.71)	180	180	0.011
2	903.88 (±722.16)	800	800	719.32 (±32.67)	700	700	0.234
3	317.63 (±112)	300	300	273.41 (±38.34)	270	270	0.069
4	293.79 (±103.67)	350	300	191.59 (±48.93)	180	180	p <0.001
5	103.44 (±48.49)	90	90	80.23 (±1.88)	80	80	0.026
6	408.74 (±161.87)	500	400	394.23 (±34.44)	400	400	0.678
7	392.21 (±116.93)	400	400	377.73 (±69.50)	320	350	0.575
8	422.16 (±142.59)	400	400	302.95 (±22.40)	300	300	p <0.001
9	733.48 (±228.61)	600	700	553.86 (±127.47)	500	500	p <0.001
10	815.50 (±299.87)	600	765	517.73 (±133.34)	450	475	p <0.001
11	707.98 (±616.86)	600	650	601.36 (±32.63)	600	600	0.420
12	652.28 (±219.19)	600	600	412.50 (±26.45)	400	400	p <0.001
13	590.19 (±207.25)	500	550	418.64 (±72.61)	350	400	p <0.001
14	672.43 (±228.99)	800	650	637.73 (±50.89)	650	645	0.481
15	658.71 (±243.31)	550	600	602.36 (±56.07)	575	576	0.282

Table 4.3 state results of calorie differences computed using descriptive metrics (mean, mode, and median) generated and ground truth calories for each meal image. Results suggest that the mode descriptive metric was the most accurate as lower calorie differences were reported which indicates that the mode was closest to the ground truth calories with an mean calorie difference of 92.73 Kcal . Mean metric calorie difference was the highest with 135.09 Kcal and mean median calorie difference was marginally higher than mean mode with 103.73 Kcal.

Table 4.3 Meal images with calorie content used in online survey.

Meal	True Kcals	Mean Kcal Difference	Mode Kcal Difference	Median Kcal Difference
Meal 1	172	80.88	28	78
Meal 2	700	203.88	100	100
Meal 3	271	46.63	29	29
Meal 4	160	133.79	190	140
Meal 5	80	23.44	10	10
Meal 6	400	8.74	100	0
Meal 7	315	77.21	85	85
Meal 8	295	127.16	105	105
Meal 9	495	238.48	105	205
Meal 10	410	405.50	190	355
Meal 11	634	73.98	-34	16
Meal 12	400	252.28	200	200
Meal 13	349	241.19	151	201
Meal 14	641	31.43	159	9
Meal 15	577	81.71	-27	23
	Average:	135.09	92.73	103.73

Table 4.4 states results of comparing crowdsourced calorie differences when compared to individual non-expert participant calorie differences. The following stages were completed in order to achieve this; (1) Mean calorie difference was computed for each individual using ground-truth calories (Table 4.1). (2) Crowdsourced mean, mode, and median calorie differences were computed using all non-expert estimations across each meal image type. Mean, mode, and median was calculated for each meal image type using all non-expert estimations. (3) The calorie difference was calculated using mean, mode, and median for each meal image with ground truth calorie. (4) The overall mean calorie difference was calculated using mean, mode, and median calorie differences generated for each meal image type. (5) This new average calorie difference generated using mean, mode, and median calorie differences were then compared to the individual non-expert mean calorie differences to determine improvement in accuracy. A lower calorie difference indicates improvement when comparing crowdsourced calorie difference to individual calorie difference..

Results show that average mode calorie difference was 92.73 Kcal and was lower than 70 participants average calorie differences, therefore outperforming 58.33% of non-expert group. Mean calorie difference outperformed 48 participants (40%) and median calorie difference outperformed 63 participants (52.5%). These results support previous research in utilising crowdsourcing to support dietary management through calorie identification.

Table 4.4 Crowdsourced calorie differences vs Individual calorie differences.

Crowdsourced Metric	Crowdsourced Kcal Difference	Participants Outperformed	Percentage (%) Outperformed
Mean	135.09	48/120	40%
Mode	92.73	70/120	58.33%
Median	103.73	63/120	52.5%
	Average:	60.33	50.28%

4.4.2 Calorie Adjustment Results

Five-fold cross-validation was used to evaluate the calorie adjustment process. In these experiments, each fold was used as a testing dataset, and the remaining were used as training to determine mean calorie differences and mean calorie estimations for each meal image. The mean calorie estimation calculated for each meal was used as a baseline, as outlined in Algorithm 1 and Figure 4.1. Figure 4.4 shows the results of original mean calorie difference and adjusted mean calorie difference for each test fold. The adjusted mean calorie difference was calculated using equation (4.2) and (4.3) for each fold and compared with original mean calorie difference for the same fold. Each fold achieves a lower mean calorie difference in comparison to the original estimations using the calorie adjustment process and the results using the rule-based system, outlined in Figure 4.1 and Algorithm 1, demonstrates that calorie accuracy improvement has been made. Figure 4.5 shows the comparison between the original mean calories, adjusted mean calories, and ground truth for each meal image. These results show that the calorie adjustment method increases the accuracy of user meal estimations by lowering average calorie estimations closer to ground truth.

Figure 4.5 depicts reduced mean calorie estimations across each meal image along with the original mean calorie estimation, and ground truth calories. Figure 4.5 reports reduced mean calorie estimations when applying the calorie adjustment method using five-fold cross-validation and the majority of adjusted mean calorie estimations are closer to the ground truth calorie. Figure 4.6 highlights the number of calories that were reduced for each meal image, this is calculated by subtracting the adjusted mean calorie value for all meal images from the mean original calorie value for all meal images. Figure 4.6 is able to highlight what meals experienced the largest mean calorie decrease when applying the calorie reduction method. Table 4.7 compares the original mean calorie estimations with the adjusted calorie mean estimations using ground truth calories for each meal image. Error percentages are

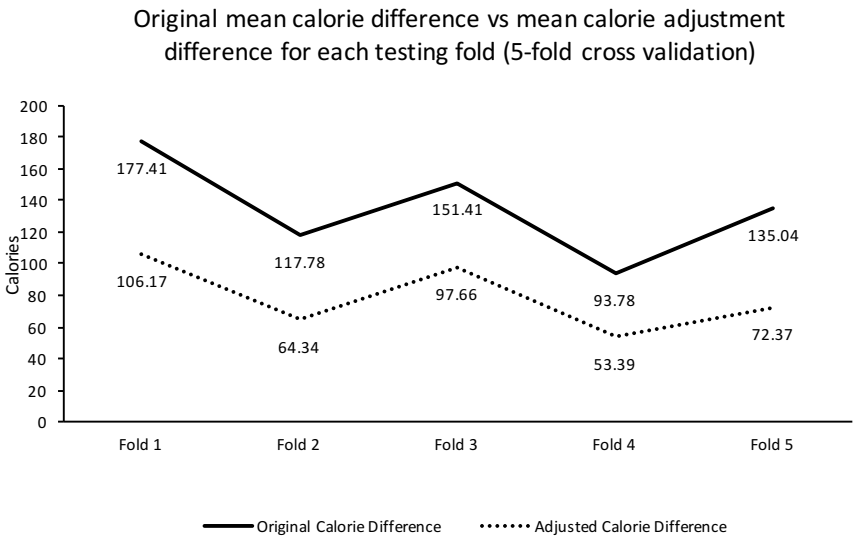


Fig. 4.4 Original mean calorie difference compared to adjusted calorie difference for each test fold.

used to assess the performance and are calculated using original mean calorie estimation, adjusted mean calorie estimation and ground truth calorie, expressed in equation (4.4).

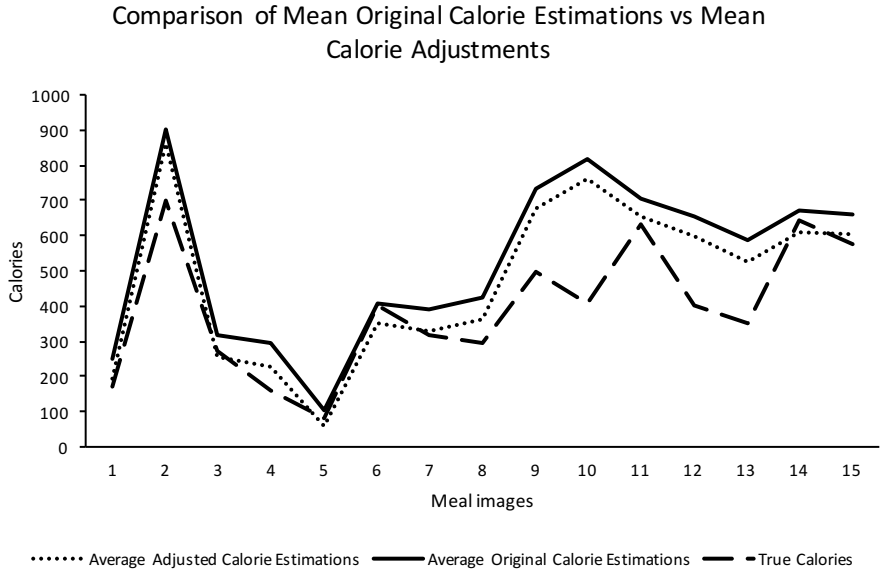


Fig. 4.5 Comparison of mean adjusted calories against mean original calories along with ground truth calories for each meal image.

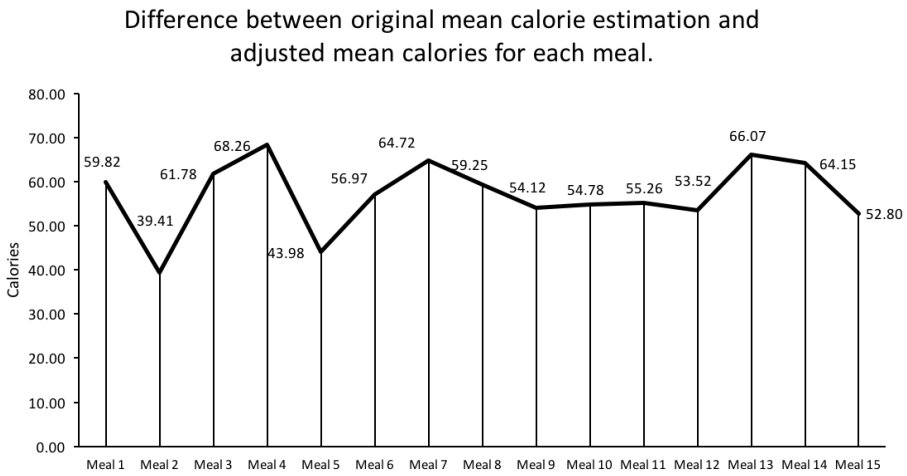


Fig. 4.6 Graph showing results of calorie deduction between mean original calorie estimations and mean calorie adjusted estimations for each meal.

Table 4.5 Identifying mean metric calorie differences estimations for each fold in 5 fold cross validation for non-expert dataset.

Meal Image	Mean Original Calorie Estimations	Mean Adjusted Calorie Estimations	Ground Truth Calorie	Percentage error Original	Percentage error Adjusted
Meal 1	252.88	193.06	172	47.03	12.24
Meal 2	903.88	864.47	700	29.13	23.50
Meal 3	317.63	255.86	271	17.21	5.59
Meal 4	293.79	225.53	160	83.62	40.96
Meal 5	103.44	59.46	80	29.30	25.68
Meal 6	408.74	351.77	400	2.19	12.06
Meal 7	392.21	327.49	315	24.51	3.97
Meal 8	422.16	362.91	295	43.10	23.02
Meal 9	733.48	679.35	495	48.18	37.24
Meal 10	815.50	760.72	410	98.90	85.54
Meal 11	707.98	652.72	634	11.67	2.95
Meal 12	652.28	598.77	400	63.07	49.69
Meal 13	590.19	524.13	349	69.11	50.18
Meal 14	672.43	608.27	641	4.90	5.11
Meal 15	658.71	605.91	577	14.16	5.01
			Average	40.85	25.52

4.5 Discussion

The aim of this work was to investigate the feasibility of utilising crowdsourcing as a method of utilising peer support and non-experts in providing calorie estimation for food logging. Research has shown that crowdsourcing nutritional

The methods presented in this work suggest that crowdsourcing may be used to adjust calorie estimations to promote accurate food logging. In regards to preliminary analysis of the collected calorie estimations, the non-expert (n=120) mean calorie estimations greatly overestimated the ground truth calories and the experts (n=22) overall mean estimates closely aligned with that of the ground truth. The analysis also showed that experts achieved high accuracy in determining calorie content of different meal images. This is evident in the descriptive statistics analysis (Table 4.2) when compared to the ground truth calories. The expert group also achieved higher accuracy in comparison to the non-expert group. Expert group mean calorie estimations were consistently accurate as shown and there was less calorie variance in the expert estimations as a low standard deviation was reported (in comparison to the non-expert group) as shown in Table 4.2. For the non-expert group, analysis shows that higher standard deviations were reported for meal images with a higher ground truth calorie, e.g. meal image 2, 9, 10, 11, 14, and 15 had the highest ground truth calories and also had the highest calorie standard deviations in the non-expert group with 722.16, 228.61, 299.87, 616.86, 228.99, and 243.31 calories respectively. Pearson correlation tests were completed on non-expert dataset using the ground truth calorie and the calorie standard deviations for each meal image. The result shows a Pearson correlation significance of 0.80 which suggests that there is a correlation between the ground truth calorie and the standard deviation of the meal images. Correlation coefficient tests were also completed using mean calorie estimations and ground truth calories for non-experts, and this resulted in a coefficient of 0.88, which also

shows statistical significance.

Descriptive statistics were generated using both expert and non-expert calorie estimations, and there was a strong correlation between the mean calorie estimations and ground truth calories for each meal image, as shown in Figure 4.2. The crowd of non-experts could collectively determine what meals had a higher number of calories. These results support other research completed in [6,7] in that crowds of individuals are able to determine healthy and non-healthy meals. T-tests were completed using calorie estimations for each meal for non-expert group and expert group, and results show that the p-values for meal image 4, 5, 8, 9, 10, 12, 13 were found to be significant with $p<0.05$ in comparing the expert mean calorie and non-expert calorie for each image. An analysis was also completed to determine what descriptive metric was most accurate in determining calories of meals in images using non-expert estimations. Results show that mode descriptive metric achieved the lowest calorie difference when compared to ground truth. Median metric's average calorie difference across all meals was slightly higher than that of mode's average calorie difference. These results indicate that a crowd of non-experts estimates can be harnessed to provide accurate calorie using descriptive statistical analysis. These results highlight the potential of using a crowd of non-experts to determine calories in food images for dietary management. Further analysis was completed that analysed the performance of individual participant calorie differences and compared them with crowdsourced calorie differences. The results indicate that crowdsourced metrics, through aggregating calorie estimations, can be used to improve calorie estimation accuracy. Results show that crowdsourced calorie estimations achieved lower calorie difference in comparison to individual participants calorie differences (Table 4.4).

Secondary experiments were completed that used crowdsourced statistical calorie metrics with the aim of adjusting calorie estimations to increase accuracy. This process was

described in Figure 4.1 and Algorithm 1. Five-fold cross-validation was used to evaluate this process. Results from these experiments show that the calorie adjustment method can reduce calorie estimations using two variables computed from each training fold; (1) overall mean calorie difference and (2) mean calorie for each meal image. This rule-based system was able to reduce individual calorie estimates to be closer to ground truth calories. Results show an overall mean error percentage reduction from 40.85% to 25.52% was achieved using the calorie adjustment method. Figure 4.6 highlights what meal images experienced the largest mean calorie reduction which was meal image 4 and 13 (2 slices of white bread toast with butter and marmalade and Porkchop, champ, and broccoli). Results of these experiments are shown in Figure 4.4, 4.5, and 4.6, and these results show that the calorie adjustment method has potential to improve calorie estimations across user food calorie predictions (results listed in Table 4.4). In regards to error percentage evaluation, meal image 4 (2 slides of white bread toast with butter and marmalade) experienced the largest calorie reduction when comparing the adjusted mean calorie estimation to the ground calorie truth amount. Meal image 7 (egg with mayonnaise and watercress sandwich 2 slices of wholemeal bread) had the lowest error percentage when comparing the adjusted mean calorie with the original mean calorie estimates. The results presented in this work suggest that crowdsourcing error overestimations can be harnessed to adjust calorie estimates to enhance accuracy for food logging.

4.6 Limitations

In regards to limitations of this work, bias is a significant issue when discussing statistical relationships between data and in order to reduce bias in this study for calorie adjustment experiments; the overall mean calorie difference was calculated for all estimations instead of using specific meal image type mean calorie differences. These experiments were

designed to minimise the bias by using a unified mean calorie difference computed using all estimations in the training folds. If specific meal image calorie differences were used (i.e. calorie difference for meal image 1) for calorie reduction, then the adjusted meal image would be biased towards that specific meal image type. Bias could be further reduced by partitioning some meal image types for training and the remaining for testing, e.g. calorie estimations for meal images 1-8, and the calorie difference could be calculated using images 1-8. The remaining images (meal images 9-15) could be allocated as a testing dataset to test the overall mean calorie difference.

In this work, a mean calorie threshold was computed for each meal image using estimations in training sets. The threshold acted as a baseline to determine if the estimation was above this threshold then the estimation would be adjusted using the overall mean calorie difference. However, in some individual calorie adjustment instances, the calorie adjustment algorithm deducted calories beyond the ground truth calorie point. To mitigate this issue, other calorie baselines could be explored instead of the mean, i.e. mode or median and to evaluate the performance of these measures. Also, exploring the use of overall mode or median calorie differences instead of using overall mean calorie differences and evaluate the performance to measure improvement, if any. More research is needed in refining the calorie adjustment process in this work through adding lower end calorie thresholds to ensure that adjusted calorie estimates do not fall below a statistical metric, i.e. mode or median as in this work some individual adjusted estimates fell below the ground truth calorie.

4.7 Key Findings

Experiments presented in this study show that crowdsourcing calorie estimates may be used to estimate calories in photographs of meals to provide dietary management. Results show that non-expert group mean calorie estimations significantly overestimated the ground

truth calories and the experts' group overall mean estimates closely aligned with that of the ground truth calories. Further analysis showed that experts achieved high accuracy in determining calorie content of different meal images. Statistical analysis showed that for the non-expert group, the study shows that higher standard deviations were reported for meal images with higher ground truth calories as results show a Pearson correlation of 0.80, which indicates that there is a correlation between the ground truth calorie and the standard deviation of the meal images (higher the calorie of a meal, the more varied the calorie estimations were from non-expert group). The analysis also suggests that crowdsourcing calorie estimations can enhance accuracy when compared to individual participant estimations. A crowdsourcing calorie adjustment algorithm was proposed, and preliminary tests were completed to show that non-expert descriptive statistics can be used to reduce mean calorie overestimations using a rule-based system to promote food logging accuracy.

4.8 Implications for Dietary Management

In regards to implications for dietary management, the results presented in this study highlight that nutritionists can accurately determine calories in meal images and thus can provide dietary and nutritional advice using food images. This approach suggests that nutritionists can accurately determine calories in meal images without taking the time to calculate calories in foods plus reducing the need to ask individuals to record the weight of food consumed. Thus reducing time and effort for both the nutritionists and individuals and making determining calories more easier. Further implications for dietary management showed that a group of non-experts could accurately assess meal images that have higher calorie content in comparison to other meal images and this echoes results in different research discussing crowdsourcing. A crowdsourcing element could be incorporated into a web application that allows users (experts and non-experts) to determine the calories of

foods from images to provide support and promote dietary management. Further experiments completed in this study also present a rule-based system for automated calorie adjustment approach, and results show that this method can reduce calorie overestimations for some food items to promote accuracy. This study suggests that crowdsourcing calorie estimation metrics of non-experts can be used to adjustment future calorie estimations to enhance accuracy.

4.9 Summary

In this study, the use of crowdsourcing was explored to predict calorie content in images of food meals. The method of using food images to document nutritional content has become popular due to the rise of smartphone usage. Users can photograph high-quality pictures of their food intake. Research was completed in ascertaining nutritional content in images through RFPM and crowdsourcing. This study aimed to explore the predictive power of 2 groups of users; experts and non-experts and results showed that a group of users were able to determine calories of meals accurately when compared to single users. The research highlighted in this study supports work completed in other research that uses crowdsourcing in determining the nutritional content of meals in images. Further analysis explored the use of crowdsourcing calorie differences to adjust calorie estimates. A rule-based system was implemented that adjusted calorie estimations if estimations exceeded a calorie threshold. Calorie thresholds and calorie deduction amounts were based on crowdsourced analysis of previous calorie estimations. Results from using a rule-based system showed that crowdsourcing has the opportunity to enhance the accuracy of calorie estimations for dietary management.

Chapter 5

Feature Fusion Food Image Classification to Support Dietary Management

5.1 Introduction

Research has shown that computer vision approaches can be used to estimate nutritional content in food images with relative accuracy as well as using computer vision approaches to predict what food items are present within a food image[175,225]. In an automated food image classification platform, the food portion is identified first, and then nutritional information is then calculated. Food logging is an essential method for dietary management, and the increasing use of smartphones has led to an increase of dietary management websites such as FitnessPal, LoseIt, and Noom Coach. For much of these applications, users are required to manually input their food items and estimate food portions. This may lead to under or overestimations. The process of manually searching a database for calorie content of food items can be cumbersome and time consuming for the user as they are required to

navigate through numerous drop down menus to identify the correct food item. Behavioural change approaches have also been incorporated into applications to promote retention rates among individuals with some success [235]. One of the aims of an automated dietary management platform is to promote usability and convenience to retain dietary management adherence. The user would be able to take capture a meal using a smartphone camera and classify the food item using computer vision approaches. Food portion segmentation can also be used before or after the food image is classified. Food image segmentation approaches can be used before to remove noise or other unrelated food items to enhance classification percentage accuracy. Food image segmentation after the food item classification to determine nutritional content through food volume measurement [236]. The number of calories could then be calculated in a food item can also be estimated by taking in account the geometric area of the food portion [237]. The food items area can be calculated using a fiducial marker such as a coin in the photograph or a shape (area of reference shape is known to the user) [237]. The food portion classification would then be used to search for the calorie content and portion size. This work presented in this chapter focuses on the classification of food portions captured in free-living environments as well as isolated (laboratory prepared food images, in which no noise is present and the image is focused on food texture image) food image textures using a variety of feature extraction approaches. Food detection models were also developed using a feature fusion approach, evaluated, and compared with other related works. The work presented in this Chapter is based on published works [238 - 241].

5.2 Aim & Objectives

- The aim of this research study was to investigate and implement feature extraction methods and machine learning algorithms to classify foods in photographs.

  The objectives of this study were:

- To identify existing food image datasets for the study.
- To assess image feature extraction approaches for food classification.
- To identify optimal image features for food classification.
- To compare machine learning algorithms for food classification.
- To identify optimal machine learning algorithm for food classification.
- To evaluate the proposed machine learning algorithm for food classification.

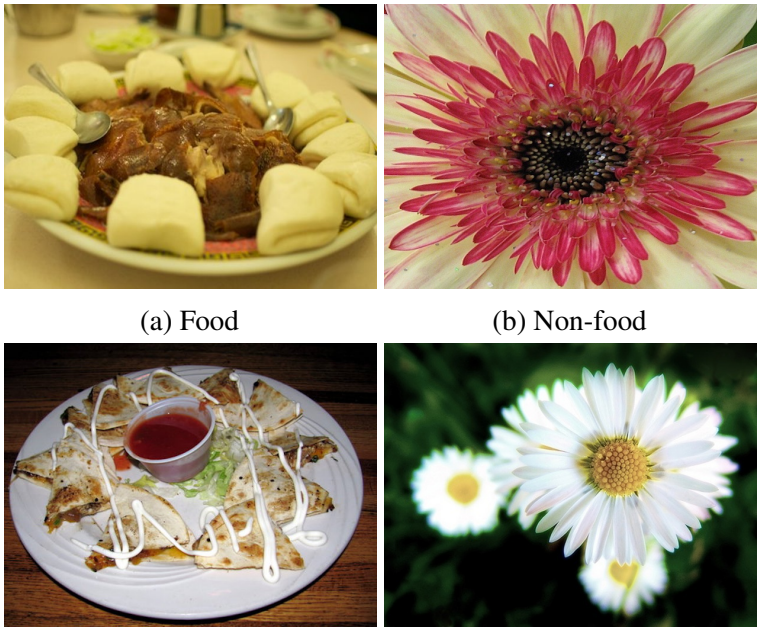
5.3 Methodology

5.3.1 Food Datasets Under Study

Three food image datasets are used in this work; Food-30, RawFooT-DB, and Food-5K. For each dataset different feature extraction methods will be used as each dataset is distinctively different. Food-30 contains images captured in free-living environments, and RawFooT-DB contains isolated food texture images that were prepared in a laboratory environment captured under different lighting conditions. Food-5K contains 2 classes, food and non-food, which contain food images captured in free-living environments. The non-food class contains images from objects and scenes that contain no food items. Figure 5.1 - 5.3 are examples of images for each food image dataset. For more information regard the food image datasets used in this work, please refer to Chapter 2 Literature Review.

Fig. 5.1 Example of food texture images in Food-30 (arepas, braised pork, bread, chasiu).

Fig. 5.2 Example of food texture images in RawFooT-DB.



(a) Food (b) Non-food

(c) Food (d) Non-food

Fig. 5.3 Images depicting food items and non-food items in Food-5K food image dataset.

5.3.2 Feature Extraction Approaches

This section discusses feature types used in this work for both Food-30 and RawFooT-DB. The feature types used in this work consist of global and local features. LAB colour space statistics will be extracted and used with bag-of-features (BoF) method to create a visual dictionary. Local features will also be extracted from the image dataset; Speed-Up-Robust-Features (SURF) will be used to extract features and a BoF model will also be applied to the SURF features to create a visual vocabulary to classify images. Segmented Fractal Textual Analysis (SFTA) and Local Binary Patterns (LBP) will also be used in this work. This section will give a brief overview of these feature extraction methods. Figure 5.4 states feature extraction approaches used and machine learning algorithms used in this study.

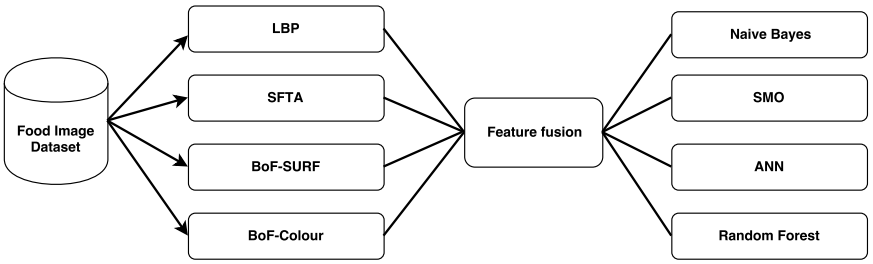


Fig. 5.4 Pipeline that states feature extraction approaches and machine learning algorithms used.

5.3.3 Bag of Features (BoF)

Bag-of-features (BoF) or bag-of-visual words (BoVW) is a technique that is used to describe an image through a series of visual word occurrences using a visual dictionary. It has been used extensively for object recognition and food image classification with research achieving high percentage accuracies, as discussed in Chapter 2 Literature Review. In this work we also used a BoF method for food image classification. The BoF technique uses a code book or a visual dictionary that is created using features extracted from the training

image set. Each visual word in the visual dictionary represents patches in a visual dictionary. An image can be classified by counting the amount of visual word occurrences that are present in the visual dictionary. The results feature vector can then be quantified using a histogram to represent the number of visual word occurrences in an image. In this work BoF was used with SURF features and LAB colour features to produce a feature vector for each image to train machine learning models.

5.3.4 Speeded-Up-Robust-Features (SURF) with BoF

For Food-30, a SURF a dense grid feature extraction approach was applied (15 pixels X 15 pixels) with a block width of [32 64 96 128]. Grid lines that intersect across each other define areas for feature extraction and block width is used to extract multi-scale SURF descriptors at each grid intersection. Initial experimentation with grid sizes yielded various results however 15 x 15 grid achieved promising results with Food-30. Food-30 is a complicated food image dataset in comparison to RawFooT-DB, therefore a more dense grid was used to extract SURF features from Food-30. For RawFooT-DB, SURF features used a grid approach also which was configured to 20 x 20 pixels with [32 64 96 128]. For Food-5K, a grid approach was also used for interest point detection was also used using 20 x 20 pixel grid using [32 64 96 128] block width.

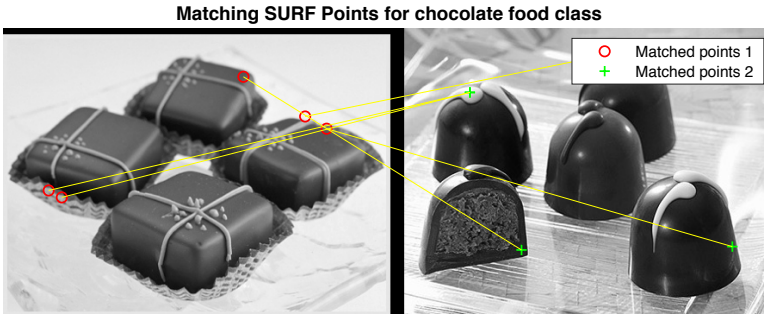


Fig. 5.5 SURF feature matching for chocolate food image class.

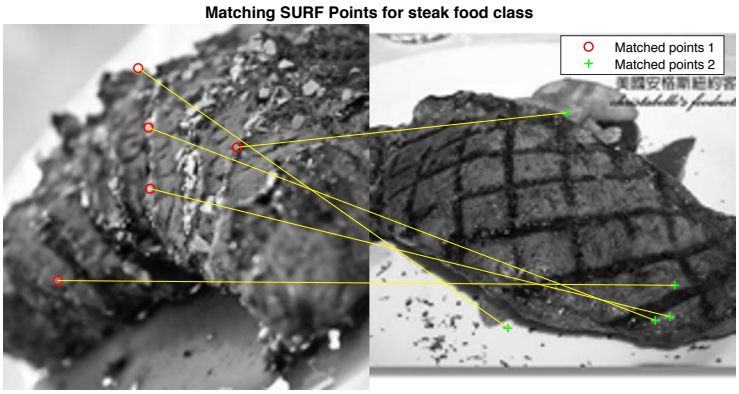


Fig. 5.6 SURF feature matching for steak food image class.

5.3.5 Segmentation Based Fractal Textual Analysis (SFTA)

Texture features were also used in this work in the form of SFTA and LBP features. SFTA features has been used for object classification works however there is little research available to applying SFTA based features for food image texture classification [95]. SFTA features were used in this work as research has shown it to be superiour in comparison to other conventional texture extraction approaches in regards to percentage accuracy and speed as discussed in [95]. For more information regarding SFTA algorithm, refer to Chapter 2 Literature Review. SFTA features were extracted from each food image dataset. For RawFooT-DB, 64 binary images were computed using different thresholds with SFTA algorithm, and from each image and features were extracted, this was done to encode greater texture as the classes in RawFooT-DB contain low in-between class variance. Figure 5.7 states the features that are extracted from each binary image and border image generated using SFTA algorithm.

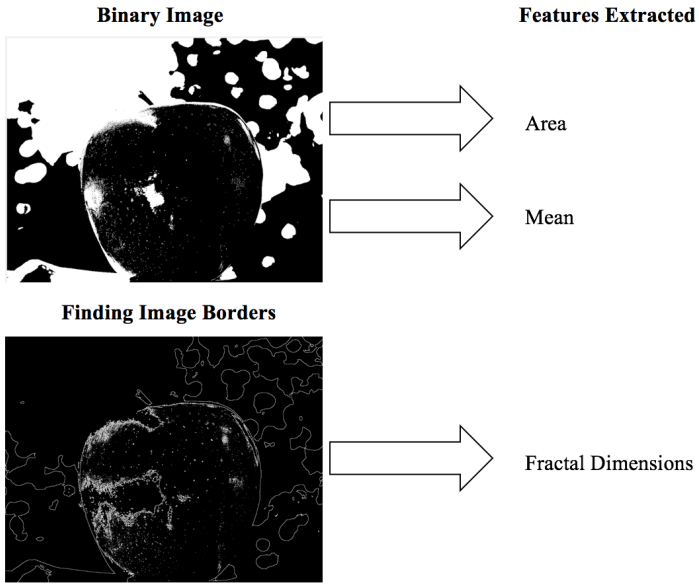


Fig. 5.7 Features extracted using SFTA algorithm.

5.3.6 Local Binary Patterns (LBP)

LBP features were extracted from Food-30, RawFooT-DB, and Food-5K. For RawFooT-DB, parameters for LBP features were manually configured to‘24’ neighbours and neighbour circular radius was set to ‘5’. Neighbours parameter is used to compute LBP for each pixel within an image. The higher the number of neighbours configured, the greater the detail encoded each pixel in an image. In regards to radius, the higher the value, the greater the detail captured over a spatial area. For Food-5K, and Food-30, LBP neighbours was also preconfigured to ‘24’ and radius to ‘5’ to extract greater detail from each pixel in an image [96]. Figure 5.8 is an example of LBP patterns highlighted within a food image from Food-30. For more information regarding LBP features refer to Chapter 2 Literature Review.

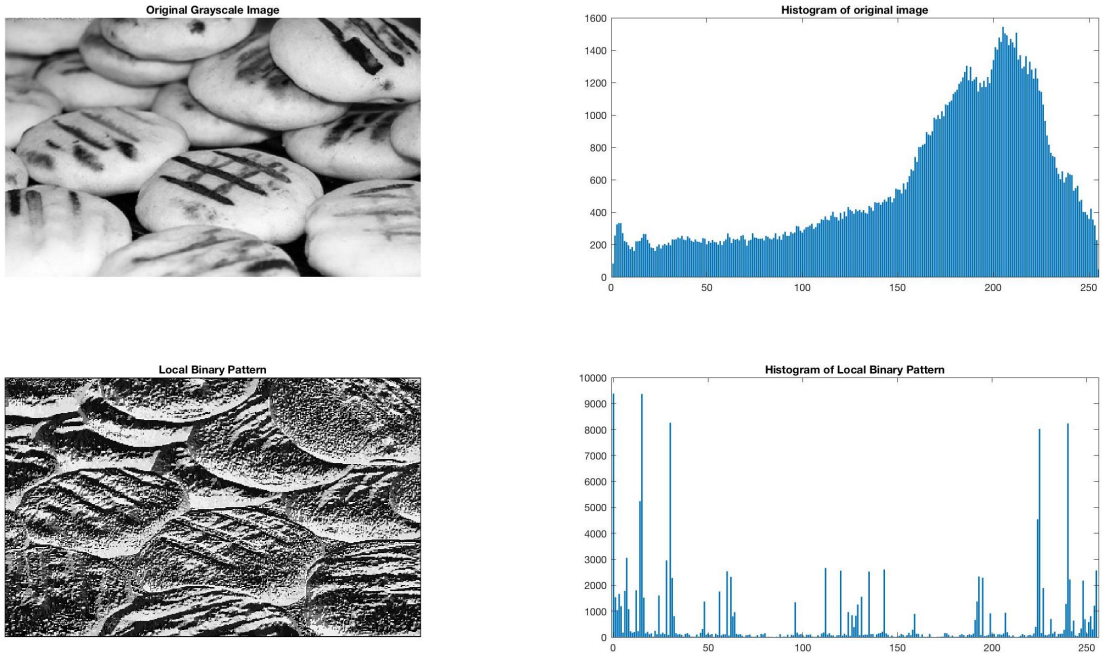


Fig. 5.8 LBP patterns being generated using [285]. Figure shows histogram of original image and same image with LBP transformation. A histogram of the LBP transformation image is generated to form a feature a vector.

5.3.7 LAB Colour Space (LAB)

LAB colour features was used in this work and used within a BoF model to create a visual dictionary to classify test images. Lab colour space is described as a 3 axis colour system; L representing lightness and A and B representing colour dimensions. There are several advantages to using LAB colour space as a method to represent colour in images; it provides a precise means of representing colour and LAB is device independent and also LAB colour space can easily be quantified to compare images. In this work, RGB images were converted to LAB colour space. Each image within each dataset used in this chapter was divided into 16 × 16 pixel blocks and the average value of each block was computed. The image was then scaled down in order to compute the average LAB colour value over the entire image. The average LAB values were then stored in a matrix and normalised using L2 normalisation. The location from where the colour feature was extracted and appended

to the feature. BoF approach was used with LAB colour space features in order to reduce the number of features extracted from each image dataset (Food-30, Food-5K). For Food-30 colour extraction, a grid approach was also used to extract greater colour detail to generalise between each class.

5.3.8 Feature Extraction

In this work, 3 distinct food images datasets are used; Food-5K, Food-30 and RawFooT-DB. Each food image dataset differs vastly in regards to image food type, composition, and variance, and because of these differences, experiments were completed that extracted different feature types from each dataset. Food-5K and Food-30 consist of images photographed in real-world environments, therefore, noise and high colour variance may exist in images. A feature combination approach was used to classify Food-30 that uses multiple feature types. RawFooT-DB is a texture based food image dataset, therefore primarily texture based features approaches; segmented fractal based features (SFTA) and LBP statistical texture features were extracted from the RawFooT-DB along with SURF features. Food-5K is similar to Food-30 is regards to composition and image acquisition, therefore a global and local feature combination approach will be used (similar to Food-30). Table 5.1 lists feature extraction approaches used for each food image dataset.

Table 5.1 Feature extraction methods used for each food image dataset.

Food-30 & Food-5K Dataset	**RawFooT-DB**
BoF-SURF	SFTA
BoF-LAB Colour	LBP
SFTA	BoF-SURF
LBP	

5.3.9 Evaluation and Statistical Analysis

Metrics were used to evaluate the performance of the machine learning algorithms. For Food-30, 10-fold cross validation was used to accurately calculate the performance metrics for each food image dataset. Confusion matrix was generated for each fold and used to generate evaluation metrics. These metrics included Cohen's Kappa statistic (equation 5.3) aswell as the mean percentage accuracy rate (number of correct classifications) as computed from each of the folds using equation 5.1. Cohen's Kappa was used to measure the agreement between the predicted class and the actual class for each food image. Initial experiments consisted of increasing the visual word count in the BoF model using 500 increments. This was done for BoF-SURF and BoF-colour for Food-30 as images in Food-30 are more complicated in comparison to RawFooT-DB and Food-5K due to the colour variation and noise. This was to completed to determine optimum visual word count for each classifier by using percentage accuracy as a measurement as well as the optimal feature combination approach. The highest percentage accuracy achieved for each classifier (using the 500 visual word increments) would be combined with the remaining feature sets extracted from the image dataset. Food-5K was partitioned into training, validation, and evaluation subsets based on original author's approach [209].

$$Acc = (TP + TN)/(TP + FP + FN + TN) \tag{5.1}$$

where TP is true positive instances, TN is true negative instances, and FP and FN are false positives and false negatives. These values are generated using the confusion matrix computed from each fold within 10-Fold Cross Validation.

$$F1 = 2TP/(2TP + FP + FN) \tag{5.2}$$

$$\kappa \equiv \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e}, \tag{5.3}$$

Similar feature extraction methods were also with Food-5K. Ten-fold cross validation was initially used with Food-5K training set to determine performance of feature extraction approaches. Food-5K training dataset was used to train machine learning classifiers and results trained food detection models were validated with Food-5K validation dataset. Food-5K Evaluation features were then used with machine learning model that achieved the highest percentage accuracy result with Food-5K validation subset. The labelled feature set combinations were extracted to a CSV file format using Matlab (R2016a and R2017b) [243] and the Weka Analysis (v3.7.13 for Food-30 and v3.8.1 for Food-5K and RawFooT-DB) [244] platform was used to train machine learning algorithms using the features extracted. For RawFooT-DB, a training and testing split was used that was created by the authors of the dataset in [20]. In this work, we followed the same image dataset partition for feature extraction, testing, and analysis.

5.3.10 Supervised Machine Learning Classifiers

In this work a range of classifiers were used to assess the performance using the extracted features types extracted. Table 5.2 is a list of the machine learning classifiers used in this work along with the parameters used for each model. The parameters in each model remained as default unless otherwise stated.

Table 5.2 High level Overview of machine learning classifiers and parameters used in this study. Learning rate was configured to adaptive for each experiment.

Machine Learning Classifiers	Parameters used
Naive Bayes (NB)	Weka default parameters
Sequential Minimum Optimisation SMO [19]	Weka default parameters
Artificial Neural Network (ANN) [24]	1 layer, 100 neurons, 1000 iterations
Random Forest (RF)	300 trees (**Food-30**) 100 iterations **(Food-5K, RawFooT-DB)**

Table 5.3 list the parameters used for each ANN model trained in this study. An adaptive learning rate is used, in which several learning rates rates are initially used and the learning rate with the lowest cost function is used to begin training. The Weka ANN plugin uses dropout regularisation to prevent overfitting and Rectified Linear Units as the activation functions [245]. Table 5.4 are the hyper parameters used for SMO for each food image dataset.

Table 5.3 Hyper-parameters used for each ANN in this work.

ANN	Parameters
Number of iterations	1000 (max)
Num of layers	1
Neurons per layer	100
Learning rate	Adaptive*
Learning momentum	0.2
Weight penalty	0.00000001 (default)
Hidden Layers drop out rate	0.5
Input layer drop out rate	0.2
Activation function	ReLu
Convergence threshold	0.2
Batch	100

Table 5.4 Hyper-parameters used for each SMO in this work.

SMO	Parameters
Fillter type	Normalise data
Kernel	PolyKernel
Random Seed	1
Tolerance parameter	0.001
C	1.0
Epsilon	1.0E-12
Calibration model	Logistic

5.4 Experimental Results

5.4.1 Food-30 Results

Various combinations of BoF-SURF and BoF-Colour were combined together with SFTA and LBP features to achieve the highest result. Table 5.5 shows the percentage accuracy of increasing the visual words using BoF-SURF and BoF-colour features for each machine learning classifier. Table 5.6 lists the experimental results of using SMO and Naive Bayes with individual LAB colour and SURF features with BoF. In Table 5.5, SMO achieves the highest percentage accuracy result with 50.27% using BoF-SURF using 2500 visual words. Naive Bayes with LAB colour features achieved the lowest percentage accuracy with 19.20%. Table 5.6 states the results of using the same visual word increments for other machine learning classifiers. ANN achieves the highest percentage results for BoF-SURF with percentage accuracy of 56.33% and Random Forest classification achieves the highest for BoF-colour features with percentage accuracy of 40.87%.

Table 5.5 Results from increasing the visual word count by 500 for SURF and colour features using BoF method. SMO classifier (SMO) and Naive Bayes (NB) was used in these experiments.(* denotes highest accuracy achieved).

Visual Words	SMO SURF	SMO Colour	NB SURF	NB Colour
500	46.30	34.67	22.73	19.20
1000	48.17	33.43	23.87	21.37
1500	48.87	34.00	24.67	21.83
2000	50.17	34.73	25.13	22.90
2500	50.27*	35.67	25.63	24.00
3000	50.27	36.43	25.47	24.40
3500	49.90	35.60	26.40*	24.53
4000	49.70	35.87	25.70	24.67
4500	49.50	36.57*	25.17	24.73
5000	49.47	35.80	25.23	25.37*

Table 5.6 Results from increasing the visual word count by 500 for SURF and colour features using BoF method. Neural Network (NN) and Random Forest (RF) classifier were used in these experiments.(* denotes highest percentage accuracy achieved).

Visual Words	NN SURF	NN Colour	RF SURF	RF Colour
500	49.80	38.67*	36.43*	40.87*
1000	52.37	37.93	35.47	40.27
1500	54.43	37.00	34.93	38.40
2000	54.33	36.33	35.07	38.23
2500	55.70	36.83	34.13	38.00
3000	54.47	35.53	33.90	38.03
3500	55.53	36.87	33.90	37.83
4000	56.33*	36.70	33.27	37.07
4500	55.97	36.37	32.80	37.33
5000	55.90	36.66	33.40	36.23

Table 5.7 Results combining different feature types. (* denotes highest percentage accuracy achieved).

Classifier	SURF+SFTA	Colour+SFTA
SMO	53.8	40.53
Naive Bayes	27.1	25.56
ANN	59.26*	44.43*
RF	40.33	43.53

The results from these experiments were incorporated into further classification tests by selecting the BoF that achieved the highest percentage accuracy in according to each feature type (SURF and colour). Further experiments were carried out by combining feature types for each machine learning algorithms. SURF and colour visual words that achieved the highest percentage accuracy were combined together for each classifier e.g. SURF and colour features that achieved the highest percentage accuracy for SMO were combined. Table 5.6 state the results of using feature combinations trained using the machine learning classification algorithms. The results from the 10-fold cross validation show that Neural Network trained with BoF-Colour, BoF-SURF, SFTA, and LBP feature combination achieved the highest percentage accuracy with 69.43%. The results using a feature combination approach,

ANN achieved the highest percentage accuracy in all feature fusion experiments.

Table 5.8 Results combining different feature types.(* denotes highest accuracy achieved).

Classifier	Colour+SURF	SURF+Colour +SFTA	SURF+Colour +SFTA+LBP	SURF+LBP
SMO	61.9	63.23	64.57	52.1
Naive Bayes	33.63	33.7	34.00	28.26
ANN	67.00*	68.73*	69.43*	60.03*
RF	48.90	50.30	48.07	37.63

Further experiments were conducted to depict the decline in accuracy when incrementally increasing the number of classes. Cohen's Kappa was noted from each experiment to measure the performance of each iteration. Figure 5.9 shows the results from these experiments. Figure 5.10 is depicts the percentage accuracy change when food classes were adding incrementally to the dataset. Figure 5.11 depicts the change in Kappa Statistic when classes were incrementally added. Results from Figure 5.9 show that SMO and ANN achieve similar percentage accuracy with ANN achieving 4.86% higher than SMO.

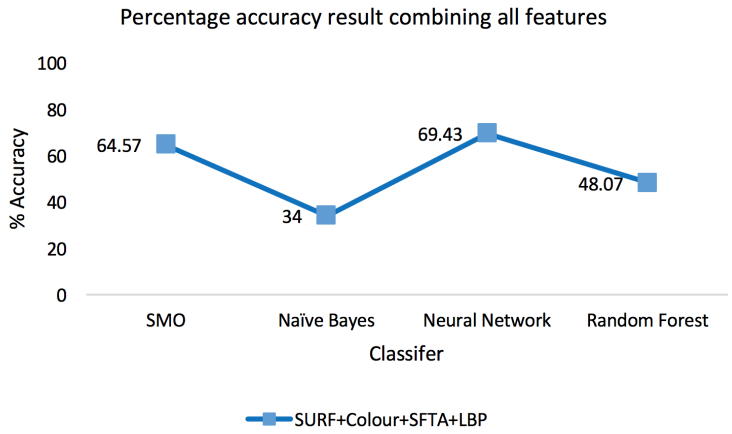


Fig. 5.9 Percentage accuracy results when combining features using different machine learning classifiers.

=

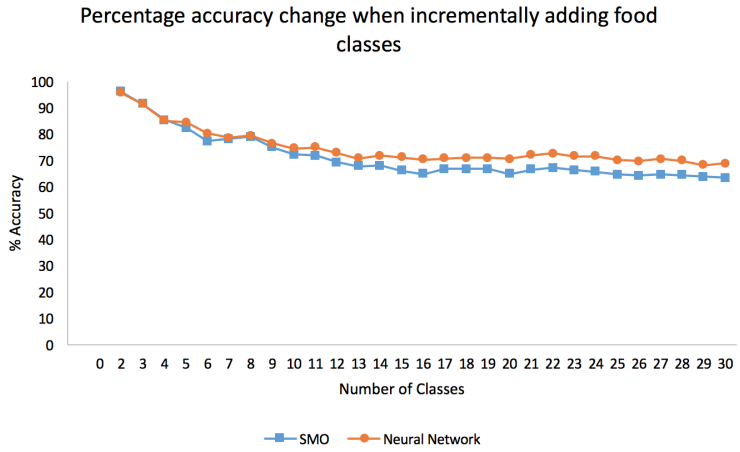


Fig. 5.10 Change in percentage accuracy when incrementally adding food classes to an image dataset. For this experiment SMO classifier was used with BoF-SURF, BoF-colour, and SFTA.

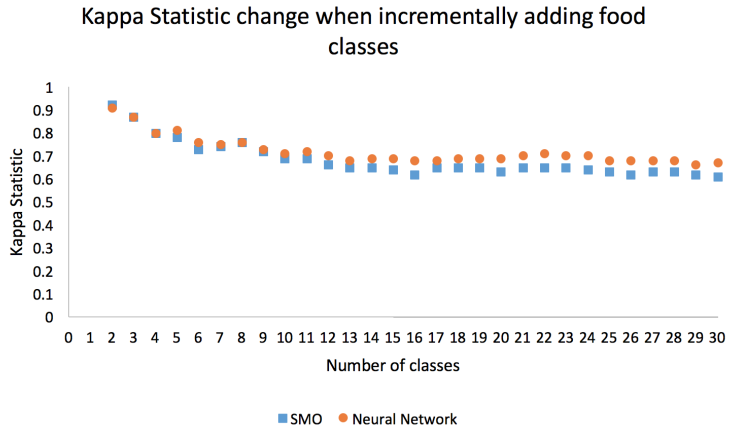


Fig. 5.11 Change in Cohen's Kappa when incrementally adding food classes to an image dataset. SMO classifier was used with BoF-SURF, BoF-colour, and SFTA

5.4.2 RawFooT-DB Results

Further experiments were carried out that used LBP, SFTA, and SURF features in classifying food texture images captured in different lighting conditions (RawFooT-DB). A testing dataset was used provided by authors of the dataset. Features were extracted from a training dataset to train machine learning models used to classify features extracted from a testing dataset. Initial experiments were completed using 10-fold cross-validation on training

dataset feature combinations. This process was done to determine what feature combinations achieve highest percentage accuracy in RawFooT-DB. Table 5.9 list the results of 10-fold cross-validation. Results using 10-fold cross-validation on various feature combinations for training set show that SMO with LBP-SFTA-SURF feature combination approach achieves the highest percentage accuracy with 99.64%. Combining SURF features with LBP-SFTA marginally increase accuracy by 1.23% when compared to the accuracy achieved with LBP-SFTA. However, when combining LBP and SFTA features together the accuracy increases 14.15% when compared to individual SFTA features. Based on these results it was anticipated that LBP-SFTA-SURF feature combination with SMO and ANN would achieve the highest accuracy when used to classify features extracted from the testing partition of RawFooT-DB dataset. Table 5.10 list the results of using various texture features extraction approaches with RawFooT-DB testing partition. Results show that LBP-SFTA-SURF feature fusion with ANN achieves highest percentage accuracy with 98.91% and SMO with LBP-SFTA-SURF features achieving second highest with 91.34%. Naive Bayes consistently achieved lowest percentage accuracy in all experiments noted in Table 5.9 and Table 5.10. Figure 5.12 - 5.14 list F-measure of each combination type and model that achieved highest overall percentage accuracy results using various feature combinations in classifying RawFooT-DB (LBP, SFTA, LBP-SFTA, and LBP-SFTA-SURF) and these results state what food image texture class achieved highest F-measure result using each feature combination set.

Table 5.9 10-fold cross validation percentage accuracy results from using texture feature extraction approaches with RawFooT-DB training dataset.(* denotes highest percentage accuracy achieved).

10-Fold CV Training Partition	SMO	ANN	NB	RF
LBP	97.83	95.97	67.1	95.25
SFTA-64	89.39	84	48.15	94.04
LBP+SFTA	99.12	98.15	77.42	98.33
LBP+SFTA+SURF	99.64*	99.38	89.20	97.89

Table 5.10 Percentage accuracy results from using texture feature extraction approaches with RawFooT-DB.(* denotes highest percentage accuracy achieved).

Testing Partition	SMO	ANN	NB	RF
LBP	74.60	82.22*	53.36	70.20
SFTA-64	77.92	75.49	44.15	77.96*
LBP+SFTA	90.25	92.38*	69.07	90.45
LBP+SFTA+SURF	93.13	93.91*	71.39	89.46

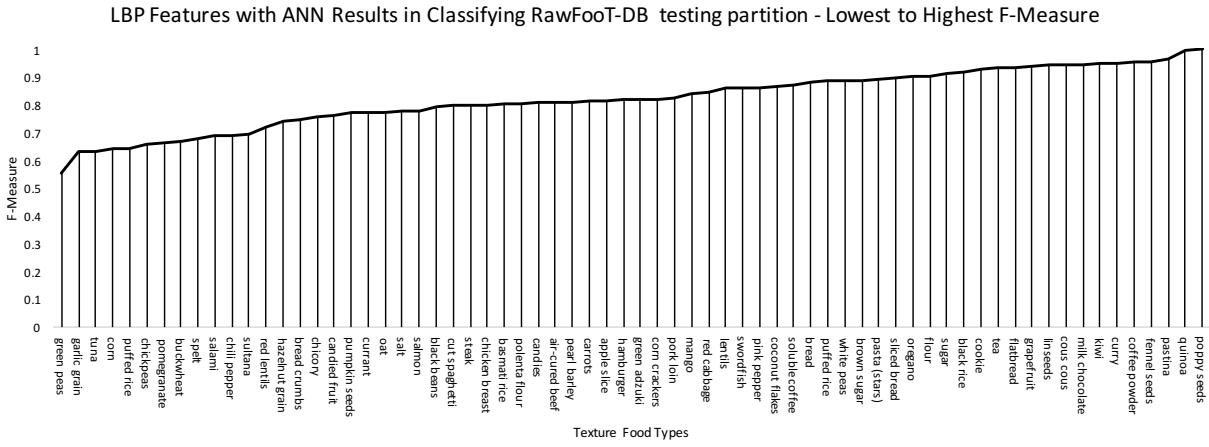


Fig. 5.12 F-Measure results using LBP features with ANN to classify RawFooT-DB.

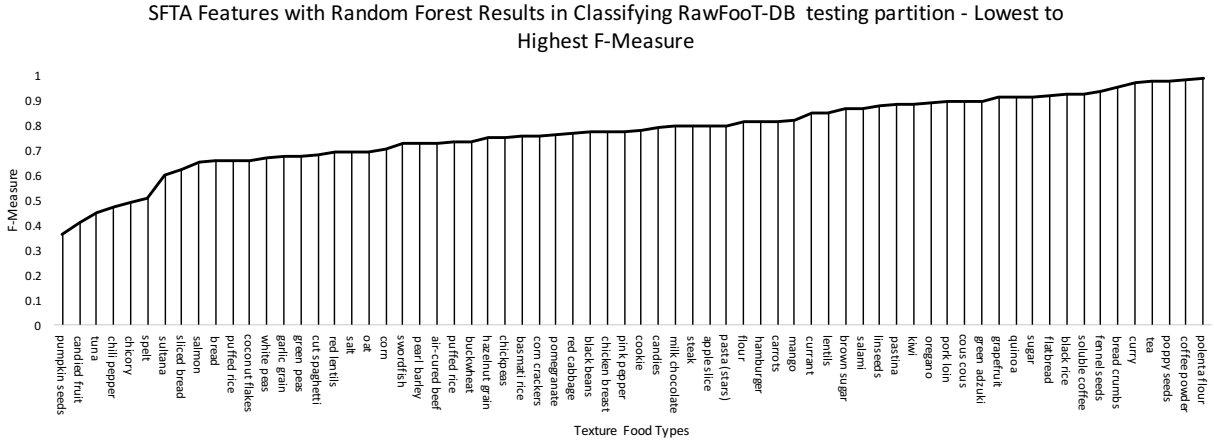


Fig. 5.13 F-Measure results using SFTA features with Random Forest to classify RawFooT-DB.

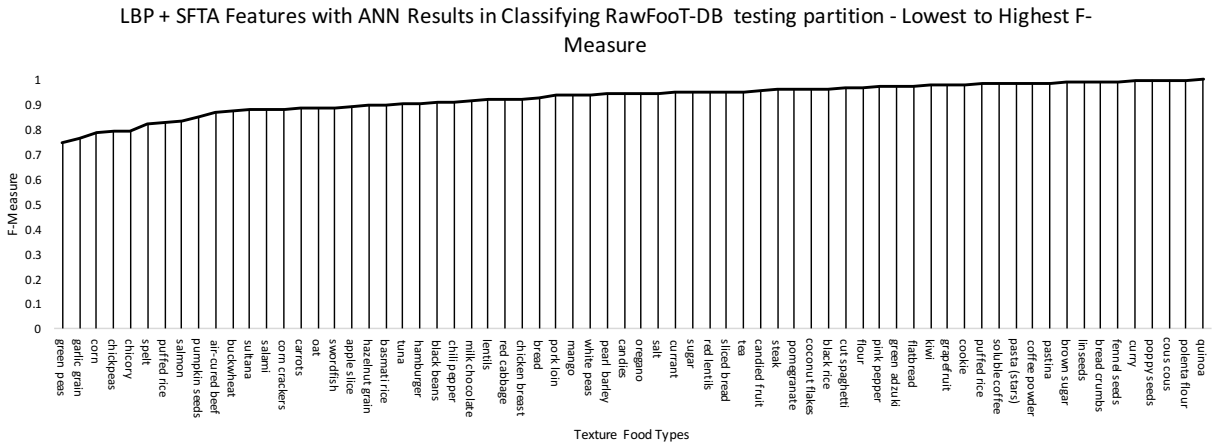


Fig. 5.14 F-Measure results using LBP and SFTA features with ANN to classify RawFooT-DB.

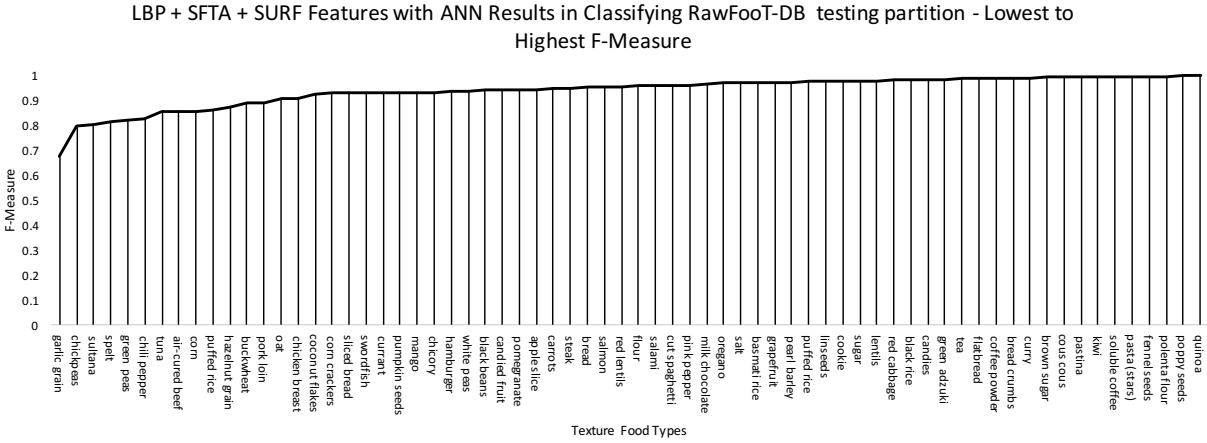
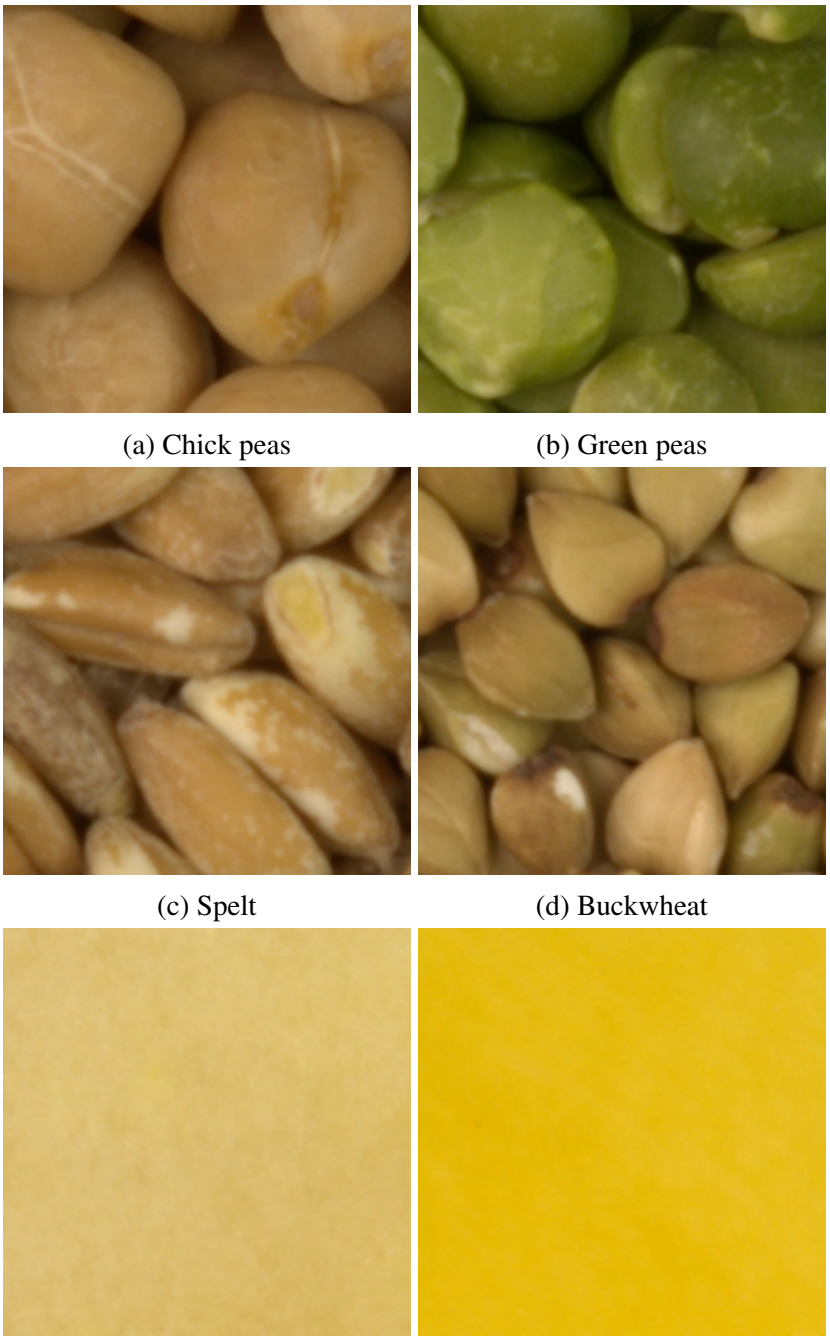


Fig. 5.15 F-Measure results using SURF, LBP, and SFTA features with ANN to classify RawFooT-DB.

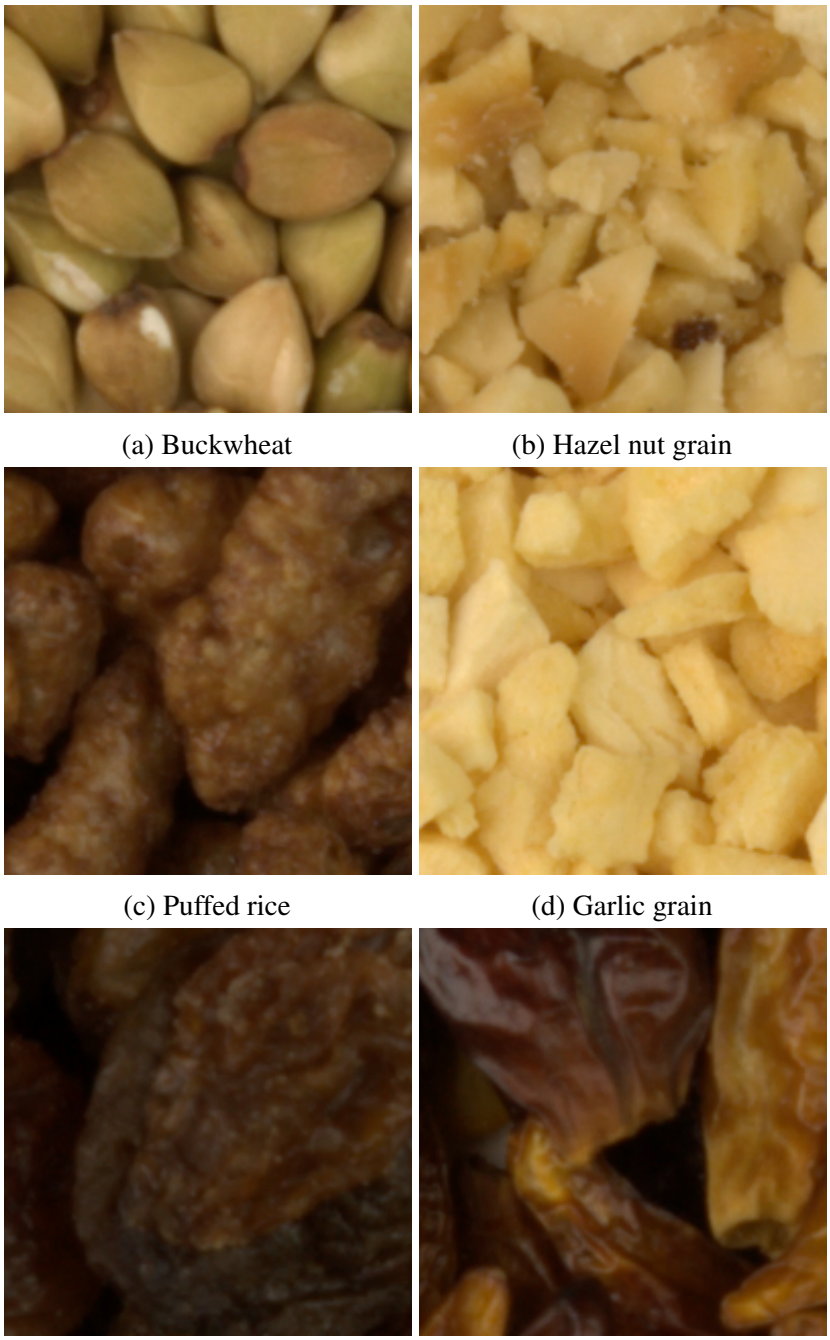


(a) Chick peas (b) Green peas

(c) Spelt (d) Buckwheat

(e) Apple slice (f) Mango slice

Fig. 5.16 Images depicting food items that were misclassified using SURF, SFTA, and LBP features with ANN. Food items on left were missclassified as items on right.



(a) Buckwheat (b) Hazel nut grain

(c) Puffed rice (d) Garlic grain

(e) Sultana (f) Chilli pepper

Fig. 5.17 Further food texture images depicting food items that were misclassified using SURF, SFTA, and LBP features with ANN. Food items on left were missclassified as items on right.

5.4.3 Food-5K Food/Non-Food Results

Experiments were completed using same feature extraction methods used for Food-30 with Food-5K. Table 5.11 list the results of single feature extraction methods with machine learning classifiers using 10-fold cross-validation on the training dataset partition in Food-5K. Table 5.12 and 5.13 list results using combined feature extraction methods with evaluation and validation datasets. Table 5.11 shows that combining all feature extraction approaches achieved highest percentage accuracy across all machine learning classifiers, therefore feature combination approach was applied to validation and evaluation datasets. Experiments revealed that ANN achieved highest percentage accuracy result in both validation and evaluation datasets with 95.8% and 91% respectively.

Table 5.11 Percentage accuracy results from using feature combinations and 10-fold cross validation with Food-5K Training Dataset.(* denotes highest accuracy achieved).

Training Dataset 10-fold CV				
Feature Type	SMO	ANN	NB	RF
BoF-SURF-500	87.33	88.8*	82.17	85.9
BoF-Color-1000	79.5	83.57*	74.83	83.37
LBP	87.33	87.63*	77.97	86.13
SFTA-32	88.63*	87.73	72.37	88.5
All features	92.37	94.13*	85.1	89.6

Table 5.12 Percentage accuracy results from using feature combinations with Food-5K Validation Dataset.(* denotes highest accuracy achieved).

Validation Dataset				
Feature Type	SMO	ANN	NB	RF
All Features	93	95.8*	85.4	91.5

Table 5.13 Percentage accuracy results from using feature combinations with Food-5K Evaluation Dataset.(* denotes highest accuracy achieved).

Evaluation Dataset				
Feature Type	SMO	ANN	NB	RF
All features	89.3	91*	82.6	86.1

5.5 Discussion

The motivation of this work was to explore the use of a combination of feature extraction methods perform in classifying 3 datasets, Food-30, RawFooT-DB, and Food-5K. Each dataset is different in terms of food image quality, lighting, and image acquisition approach. Food-30 consisted of images from the real world and free-living environments and therefore suffers from a considerable amount of noise and high colour variance. Some of the images in Food-30 are not 'cleaned' and may contact multiple food objects and non-food related items. RawFooT-DB consists of 68 food texture classes from varied foods, and each class consists of different image patches of these food textures photographed under different lighting conditions. In the experiments presented in this chapter, different combinations of feature extraction approaches were used for each dataset. Food-30 is a difficult dataset in comparison to other due to how images were developed, noise, colour variation, and quality and contains 30 food classes from a wide variety of food types. The size of this dataset coupled with image variations makes it difficult in comparison to RawFooT-DB and Food-5K. Therefore for Food-30 multiple feature types were combined to determine optimal feature combination. Experiments were carried out using BoF approach to determine the optimal visual vocabulary size for classifying Food-30. BoF fusion approach was used by combining a BoF SURF with BoF LAB colour features. LBP features and SFTA features were also

used to enhance accuracy. The performance of using feature extraction methods and machine learning classifiers with Food-30 was assessed using a 10-fold cross-validation approach.

Results show that ANN trained with BoF SURF, BoF-Colour, SFTA, and LBP achieves the highest accuracy with 69.43% accuracy. ANN consistently achieved higher accuracy across all feature combinations, and Naïve Bayes achieved the lowest accuracy in each feature combination experiments for Food-30. Table 5.14 is a comparison table from other works completed along with the accuracy and feature types; this shows that the results achieved in this work are comparable with other results achieved in this area. Table 5.14 also highlights results achieved using other feature extraction approaches used in this Chapter. From the results listed in Table 5.14, it is clear that a feature fusion approach is needed to enhance the percentage accuracy. ANN consistently achieves the highest percentage accuracy across all feature extraction approaches.

Table 5.14 Table showing comparison with other related works. Bold highlights highest percentage accuracy achieved in this work.

Method	% Accuracy
SVM + mixture of global and local features [274]	0.861 (average across all tests, 39 classes)
BoF/Fisher-Vector [166]	79.2% (top 5 classes)
Random Forest for component mining [213]	50.76 (101 classes)
Texture Features + Neural Networks [275]	79.2% (segmented food images, 46 classes).
ANN- All features*	**69.43%** (30 classes, unsegmented images)
ANN - SURF & SFTA	59.26% (30 classes, unsegmented images)
ANN - Colour & SFTA	44.43% (30 classes, unsegmented images)
ANN - SURF	56.33% (30 classes, unsegmented images)

It is important to note that the images used in this work were not segmented, but the entire image was used for feature selection. From the experiments, it is revealed that an

accuracy of 69.43% can be achieved through classifying non-segmented meal images by utilising a feature combination approach. This could be increased by segmenting the meals to promote feature selection accuracy and ultimately classification accuracy. The methods used in classifying Food-30 could be used with the methods presented in Chapter 3 in applying a user-led segmentation tool. This would allow the user to draw around the food portion using a polygonal segmentation tool. Some computer vision based automatic segmentation methods had some success in related research however due to the complicated nature of some food portions and the vast majority of food types, some user interaction may be needed, e.g. GrabCut segmentation method [118]. Therefore a user-led segmentation based approach could be incorporated with feature fusion methods presented to accurately segment and classify the food portion for nutritional content calculation.

For RawFooT-DB, a training and testing split was provided by the datasets' authors [259]. From experiments, results reveal that it was the combination of SURF, SFTA, and LBP achieved the highest accuracy with 93.91% using ANN model. Results show that quinoa class achieved highest F-measure with 1 using LBP, SFTA-SURF combination approach, stated in Figure 5.14. Garlic grain texture food image class achieved the lowest F-measure with 0.67, however, it is interesting to note that garlic grain achieved a higher F-measure in using LBP, SFTA with ANN. Garlic grain was incorrectly classified as puffed rice for several images, and several image classes were incorrectly classified as garlic grain such as red lentils, hazelnut grain, puffed rice, pearl barley, and oat Figure 5.18). Other classes were misclassified using LBP-SFTA-SURF with ANN approach using ANN, for example, 13 chickpeas texture images were classified as green peas. Further inspection shows that chickpeas images share characteristics with green peas in regards to shape, texture, and size shown in Figure 5.12. However, LBP-SFTA-SURF with ANN approach was still able to achieve 0.80 F-Measure in classifying chickpeas images. Table 5.15 lists the classes that achieved the lowest F-measure based on each model that achieved the highest accuracy for

each feature combination approach. It is clear from Table 5.15 that many of the food classes that achieved lowest F-measure across all feature combination approaches share similar characteristics in that the majority are kernel/seeds based food items with low in-between class variance.

Table 5.15 Food classes with lowest F-measure for each feature combination approach with RawFooT-DB.

RawFooT-DB Model	**10 lowest classes (based on F-measure)**
LBP + **ANN**	green peas, garlic grain, buckwheat,pomegranate, spelt
	salami, corn, chili pepper, sultana, puffed rice
SFTA + **RF**	tuna, pumpkin seeds, sultana, spelt, garlic grain
	green peas, chili pepper, chicory, puffed rice
LBP + SFTA +	green peas, garlic grain, corn, chickpeas, chicory, candied fruit
ANN	spelt, puffed rice, salmon, pumpkin seeds, air-cured beef
LBP + SFTA +	garlic grain, chickpeas, sultana, spelt, green peas
SURF + **ANN**	chili pepper, tuna, air-cured beef, corn, puffed rice

Results also reveal that using conventional based feature extraction approaches (LBP-SFTA-SURF) with ANN achieved high results in classifying food textures with low in-between class variance as these food classes are visually very similar. For example, apple slice texture images and mango slice texture images are very similar (Figure 5.16, (e) and (f), however apple slice F-measure was 0.94 and mango was 0.93. Even without the use of SURF features, LBP-SFTA with ANN achieved 0.89 F-measure in classifying apple slices, and mango achieved better without SURF features with 0.94 F-measure (Figure 5.16). Similar with spelt and buckwheat classes (Figure 5.16 (c) and (d), both texture types

share characteristics in shape, and local texture patterns however spelt class achieve 0.82 F-Measure and buckwheat achieved 0.89 F-Measure. It was also observed that LBP achieved higher results than SFTA in all machine learning classifiers except for SMO, however when combined LBP-SFTA, accuracy increases across all machine learning classifiers. Figure 5.16 - 5.17 are example classes that were misclassified due to having low in-between class variance using LBP-SFTA-SURF with ANN.

Table 5.15 list the food classes that achieved lowest F-measure for each food image texture classification model and several classes repeatedly appear lowest F-measure classification for each feature combination approach used, i.e. green peas, garlic grain, chickpeas, corn, and spelt. Further investigation shows that these food texture classes share similar characteristics as they are kernel-based foods (seed/grain) and therefore similar texture and shape patterns are shared across these food textures. Several meat classes also repeatedly appear in 10 lowest F-measure food classes, e.g. salami, tuna, salmon, and air cured beef. Further investigation into these food types shows that each share similar characteristics with other meat classes that could lead to misclassification. Figure 5.18-5.21 are examples of various meat food texture classes that were misclassified with other food texture classes.

Fig. 5.18 Food classes misclassified as **tuna** using ANN with LBP, SFTA, and SURF (red cabbage, crackers, puffed rice).

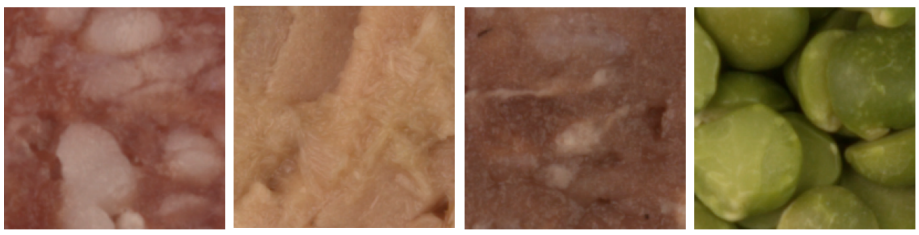


Fig. 5.19 Food classes misclassified as **salami** using ANN with LBP (tuna, hamburger, green peas)

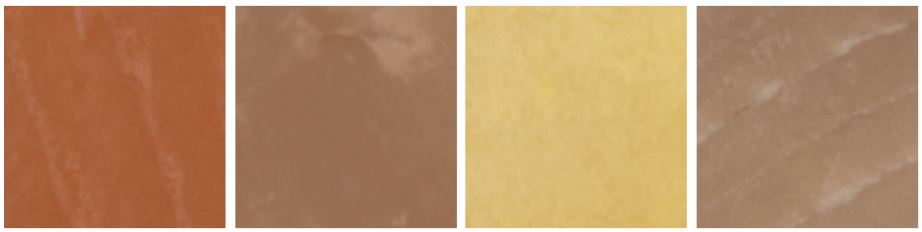


Fig. 5.20 Food classes misclassified as **salmon** using ANN with LBP and SFTA (chicken breast, apple slice, sword fish)

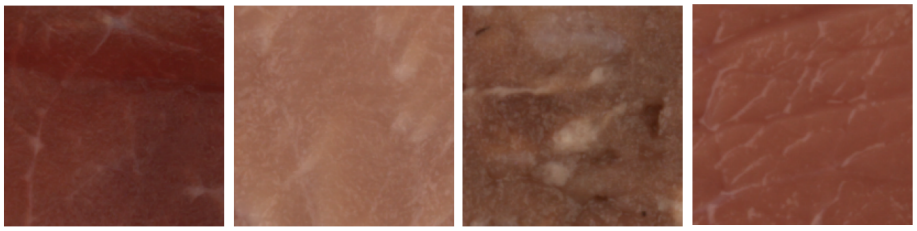


Fig. 5.21 Food classes misclassified as **air-cured beef** using ANN with LBP and SFTA (pork loin, hamburger, steak)

(a) Garlic grain (b) Hazel nut grain

(c) Chicory (d) Pumpkin seeds

Fig. 5.22 Images depicting food items that were misclassified using SURF, SFTA, and LBP features with ANN. Food items on left were missclassified as items on right.

It is clear from results that the combination of LBP-SFTA-SURF with ANN and SMO models are able to achieve percentage accuracy of 93.91% in classifying isolated texture patches under different illuminations. The results presented in this chapter, in regards to RawFooT-DB, achieves similar results in previous works using the same food image texture dataset with other conventional feature extraction techniques. Table 5.16 is a collection of some of the results from [259] and work presented in this chapter achieves similar results using SFTA based features combined with LBP and SURF. It is clear that a feature combination approach for conventional based feature extraction types is needed in classifying both real world food image datasets and datasets will low in-between variance such as RawFooT-DB

and also for food detection (Food-5K). The results presented in this work also support the need for an efficient segmentation based methods (automatic or user-led segmentation) to isolate relevant food portions for accurate classification.

Table 5.16 compares the performance of the results achieved in this Chapter with results authors of RawFooT-DB achieved [259]. Results show that a feature fusion approach is needed to increase the classification percentage accuracy when comparing single feature result types reported in this Chapter and in other works [259].

Table 5.16 Results comparison of other works using feature types to **RawFooT-DB**. The results achieved in this Chapter are noted with *.

Feature Type	Result
Hist. L	78.32%
Hist. H V	96.38%
Hist. RGB	94.9%
BOVW	89.73%
HOG	46.74%
Gabor RGB	93.02%
LBP+SFTA+SURF(ANN)*	93.91%*
LBP+SFTA (ANN)*	92.38%*
SFTA (RF)*	77.96%*
LBP (ANN)*	82.22%*

Food detection is an important step in any automated food image dietary management system. In this study, a combination of features were investigated and used with machine learning algorithms to detect food images. BoF with SURF, BoF with colour, SFTA, and LBP were used individually and combined together with each machine learning algorithm and

results show from initial experimentation with training dataset using 10-fold cross validation show that using all features combined achieved highest percentage accuracy with 94.13% using an ANN. In regards to single feature types, SURF features with ANN and SFTA features with SMO achieved highest percentage accuracy with 88.63% and 88.8%. Validation and evaluation datasets achieved 95.8% and 91% accuracy respectively. All single feature types with each machine learning classifier achieved over 70% accuracy and experiments are promising in using conventional feature training for food image detection. Comparing individual feature experiments and combination feature approach, there is a small but significant difference between the highest accuracy achieved (Table 5.12 - 5.13). Table 5.17 shows comparison of Food-5K results achieved in this Chapter with other works that also used Food-5K dataset. Food-5K is a relatively new dataset and comparative works have used CNN based approaches. More information regarding CNN is presented in Chapter 2 and Chapter 6 of this thesis.

Table 5.17 Results comparison for other works using **Food-5K**. The results achieved in this study are noted with *.

Feature Type	Result
GoogleNet (Fine-tuned CNN) [209]	99.2%
ResNet-152 (Chapter 6 Deep CNN Feature Extraction)	98.8%
All features (This Chapter, Evaluation dataset)*	91%

5.6 Key Findings

This chapter evaluated machine learning, and feature extraction approaches for food image classification and detection. Results showed that SMO and ANN machine learning

classifiers achieve highest percentage accuracies in classifying food images in free-living environments (Food-30). ANN and SMO achieve highest percentage accuracy in classifying RawFooT-DB texture images using a combination of SURF, LBP, and SFTA features. For Food-5K, ANN and SMO also achieved highest percentage accuracies using a feature combination approach. The combination of conventional feature extraction approaches (SURF, colour features, SFTA, LBP) achieves high percentage accuracies with machine learning algorithms for food detection (ANN and SMO). The combination of LBP, SFTA, and SURF can accurately identify food texture classes with low in-between class variance, and results are similar with state of the art approaches (CNN based features). The experiments presented in this chapter highlight the performance capability of conventional feature extraction approaches when used with isolated food texture images. LBP, SFTA, and SURF features can accurately classify isolated food texture images captured under different illuminations, and results using RawFooT-DB indicate that it is important to remove unnecessary noise from images and isolate food texture to achieve efficient performance. Experiments with RawFooT-DB also highlight the potential of using conventional texture feature extraction approaches to achieve high accuracy in classifying food with low in-between variance. Results achieved with RawFooT-DB are comparable with other feature extraction approaches (Table 5.16). This research suggests that using a conventional feature combination approach is needed in achieving high accuracies across image types acquired through different modalities.

5.7 Implications for Dietary Management

Results presented in this study show that conventional feature extraction approach can achieve percentage accuracy results of 69.41&, 93.91%, and 91% in classifying Food-30, RawFooT-DB, and Food-5K food datasets when feature fusion approach is applied. Results from Food-30 dataset suggest that conventional image extraction approaches coupled with

machine learning algorithms can classify food images in free-living environments. Results using Food-30 were promising, and these feature extraction methods and classification approaches could be combined with the image segmentation methods proposed in Chapter 3 'Semi-Automated Estimation of Calories of Meals in Photographs' for dietary management. Experiments also utilised texture features for classifying food texture images under different illuminations, and results show that SFTA, LBP, and SURF features used with ANN can efficiently generalise between different food textures and achieves comparable percentage accuracy to other feature extraction approaches used [259]. These results show that using conventional texture features can classify isolated food texture images, and should be combined with a manual segmentation approach proposed in Chapter 3. Other experiments used feature fusion approaches with machine learning algorithms for food image detection. Results show that conventional features can efficiently detect food in images captured in free-living environments and results are comparable with research that utilises deep learning approaches for the same task. These results for RawFooT-DB experiments show that using conventional texture features can classify isolated food texture images, and should be combined with a manual segmentation approach proposed in Chapter 3.

5.8 Summary

The work presented in this chapter focused on classifying features extracted from 3 different datasets, Food-30, Food-5K, and RawFooT-DB. Food-30 consists of images taken in real-world environments, and RawFooT-DB is a texture image dataset consisting of food textures photographed under different lighting conditions. A variety of feature extraction methods were used for each dataset due to each datasets composition and acquisition method; a variety of feature extraction methods were combined for Food-30, and texture feature methods were used for RawFooT-DB. For Food-30, experiments were completed to determine

optimal visual vocabulary for SURF and colour features with BoF approach as Food-30 images contains images from a wide variety of food types and acquisition methods. Much of the images were captured in free-living environments; therefore, it was much more complicated in comparison to Food-5K and RawFooT-DB. For RawFooT-DB a combination of different texture features was used due to the RawFooT-DB composition (isolated textures). Food-5K dataset composition is very similar to Food-30 as images were captured in free-living environments and contain noise and multiple food items, however, Food-5K is a binary dataset containing food and non-food images and the same features extracted from Food-30 were extracted from Food-5K. Experiments were completed for each dataset using single feature types and combining feature types. Results showed that a combination approach achieves a higher accuracy result for both Food-30, Food-5K, and RawFooT-DB. The results in this work are similar to results in previous food image classification works [209, 246] in that a feature fusion approach enhances the classification performance. For RawFooT-DB, an ANN trained using LBP, SFTA, and SURF features achieved 93.91% accuracy, these results achieved are similar to other results achieved using the same dataset (Table 5.16). RawFooT-DB experimental results indicate that conventional based feature extraction approaches can be used to classify isolated food texture images captured under different illuminations with high accuracy. These results also indicate that for accurate food image classification using conventional based feature extraction approaches, image segmentation is required to 'clean' the image of noise or unrelated food items. Further experiments were completed using Food-5K image dataset using BoF-SURF, BoF-Colour, SFTA, and LBP features. Results show that a feature fusion approach achieves highest percentage accuracy with 95.8% using ANN with Food-5K validation dataset and 91% using Food-5K evaluation dataset and that using texture-based features with SURF, and colour features can detect food items in images with an accuracy of 93.91% accuracy. The results with Food-5K indicate that a feature fusion

approach is needed as percentage accuracy increases from 77.9% for SFTA with Random Forest to 93.01% with ANN when using SFTA, LBP, and SURF features combined.

Chapter 6

Combining Deep Residual Features with Supervised Machine Learning Classifiers to Classify Food Image Datasets

6.1 Introduction

The increase in smartphone usage and portable technologies has also led to the rise of well-being applications that can facilitate food logging. As discussed in previous studies, many of these applications incorporate a simple diary entry and connect to an online database/API to search for nutritional content for each of the user's entries. Other methods allow users to use photographs for food logging to ascertain accurate food portion estimations and more recent technologies utilise computer vision approaches to develop automatic food logging systems to determine food item, and portion size in photographs and studies have been published in utilising computer vision methods to classify photographs of food to promote food logging with some success [236, 237, 238]. Other studies utilise conventional based approaches for automated food logging with some success, however, the popularity of

deep learning CNN based approaches have outperformed traditional methods in regards to accuracy. To train a CNN 100,000 of images are needed with a graphical processing unit (GPU), and this process may take days or weeks depending on the CNN architecture and capabilities of the GPU. However, for many image classification problems, training a CNN from scratch may not be feasible due to the amount of image data available/needed, or the processing power is not available, therefore fine-tuning, or deep feature extraction may be used. This study investigates the use of state of the art pretrained CNN architectures that were trained using ImageNet Large Scale Visual Recognition Competition (ILSVRC) dataset for deep feature extraction across various food image datasets to inform the development of food logging applications for dietary management. The remainder of this study is structured as follows; Section 6.2 states the aim and objectives of this study, part 6.3 discusses related work in how this problem has been tackled in previous research. Section 6.3 addresses the aim, objectives, and contributions of this work. Section 6.4 discusses the food image datasets used with CNN for feature extraction. Section 6.5 details the methods used in this study. Section 6.6 discusses the experiment results. Section 6.7 discusses the main findings from the experiments. The work presented in this Chapter is based on published works [246].

6.2 Aim & Objectives

- The aim of this study was to utilise state-of-the-art deep learning algorithms for feature extraction in food image classification.

  The objectives of this study were:

  - To compare state of the art pretrained CNN for deep feature extraction.
  - To apply deep features to machine learning classifiers for food image classification.

- To determine the best performed machine learning classifier for food image classification.
- To evaluate the proposed machine learning classifiers and deep features for food image classification.

A series of experiments were completed that used the features extracted from CNNs and used them as input into conventional machine learning algorithms. To answer these research questions a number of objectives needed to be completed to achieve the aim of this work; (a) a number of food image datasets needed to be selected, (b) several pretrained CNNs also needed to be identified from the literature for deep feature extraction, (c) supervised machine learning algorithms also needed to be identified to classify the images using the extracted deep activations (d) statistical analysis is then applied to the results to evaluate the methods used. The next section will discuss in detail the methods used in this work.

6.3 Methodology

6.3.1 Food Image Datasets Under Study

In this work we identified publicly available food image datasets to use for the experiments to determine efficiency of using pretrained CNNs to extract deep features for image classification. Four image datasets were used in this work. Refer to Chapter 2 Literature Review for discussion on these food image datasets.

1. Food-5K
2. Food-11
3. RawFooT DB
4. Food-101

5. UNICT-889

6. Caltech-101

Table 6.1 Table showing name, number of categories, images per category, as well as how the image datasets were developed of each food image dataset.

Dataset	Catergories	Images Per Catergory	Image Preparation
Food-5K [209]	2	1500 (training set) 500 (val & eval sets)	Real world
Food-11 [209]	11	Unbalanced	Real world
RawFooT DB [259]	68	368 each in training/test set	Controlled
Food-101 [213]	101	1000	Real world
UNICT-889 [260]	889	Unbalanced	Real world
Caltech-101 [261]	101	Unbalanced	Mixture

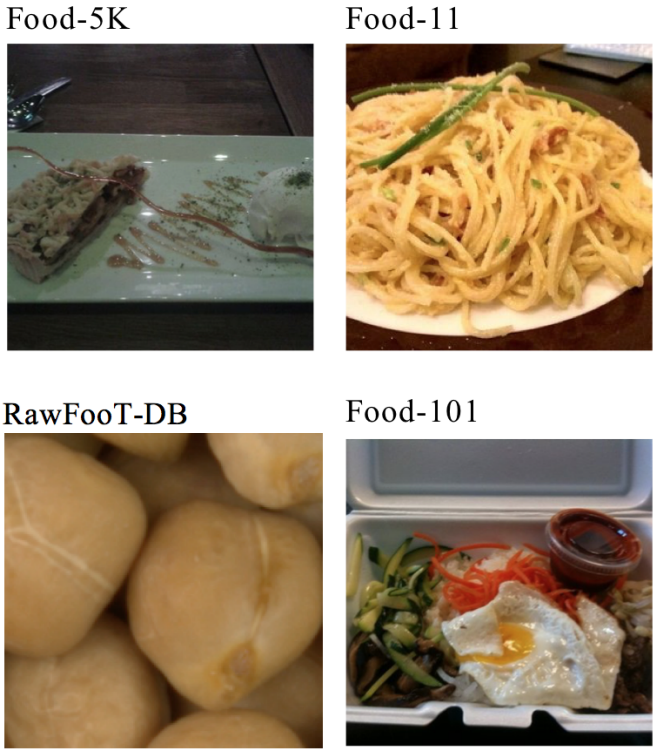


Fig. 6.1 Example of images from 4 food image datasets used in this work.

6.3.2 Datasets for Evaluation of Food/Non-Food Detection Models

Due to the small size of Food-5K, two other datasets have been used to evaluate our trained food/non-food models; UNICT-FD889, which is a food image dataset, and Caltech, which is a non-food image dataset. Deep features were extracted from UNICT-FD889 and Caltech and classified by models that achieved the best performance in classifying Food-5K datasets.

Fig. 6.2 Example of images contained in UNICT-FD889 dataset.

6.3.2.1 UNICT-FD889

UNICT-FD889 (Figure 6.3) was used to evaluate food/non-food models trained using Food-5K [53]. UNICT-FD889 contains 889 distinct food dishes to study food representation and the images are photographed in real world environments which means that much of the images may contain high food variance, however the images in UNICT-FD889 contain images that are focused on the food item with little noise.

Fig. 6.3 Example of images contained in UNICT-FD889 dataset.

6.3.2.2 Caltech-101

Caltech-101 dataset (Figure 6.4) was also used for evaluating food/non-food classification models. Caltech-101 contains 101 image categories and each contains between

50-800 images. The categories are non-food based and contain images relating to animals and objects and each image is around 300x200 pixels in size [261].

Fig. 6.4 Example of images contained in Caltech-101 dataset.

Table 6.2 Table showing testing methods used for each food image dataset. * denotes dataset splits supplied by dataset authors.

Dataset	Dataset Partition
Food-5K	Training, validation & evaluation*
Food-11	Training, validation & evaluation*
RawFooT DB	Training & testing*
Food-101	75:25 training & testing
UNICT-FD889	Testing
Caltech-101	Testing

6.3.3 UNICT-FD889 & Caltech-101 Food/ Non-Food Dataset

As well as using the validation and evaluation datasets supplied with Food-5K, further evaluation was completed with UNICT-FD889 dataset and Caltech-101 dataset in detecting food images. UNICT-FD889 is a food dataset containing images from a range of food types and Caltech-101 is a non-food image dataset. These 2 datasets were combined to create a new food/non-food dataset called UNICT-Caltech to evaluate our food detection models. Deep features were extracted from the new food/non-food dataset. Further evaluation was

completed because Food-5K evaluation and validation datasets are small with only 500 images in each category for each dataset. Using another larger dataset for evaluation can give a stronger performance indication of our models in classifying a large variety of food and non-food images.

6.3.4 Overview of Convolutional Neural Networks

The use of CNN gives great potential for applying them to a variety of problem areas ranging from food image classification to other health informatics areas [263, 264]. CNNs have been applied to various image classification problems, which have reported increase accuracy in comparison to conventional feature extraction approaches (i.e. SURF, SIFT), as discussed in Chapter 2. Research have used CNN in various ways, training from scratch, fine-tuning CNNs, or CNN for deep feature extraction. These approaches depend on the problem area and dataset size and computational power. Chapter 2 of this thesis describes in detail the CNN approaches used for image classification. In this Chapter, deep feature extraction will be used with various public food image datasets to train conventional machine learning algorithms for food image classification.

6.3.5 Image Preprocessing for Feature Extraction

The pretrained CNNs used in this work were trained specifically with requirements placed on the input images. Therefore, in order to extract deep feature representations of these images using these CNNs, it was important to ensure that the images meet the same requirements. The first requirement was to ensure that the images were resized to a specific height and width configured in the image input layer of the pretrained CNN. The images are also normalised and this is achieved by subtracting the mean of the image. The mean is removed from the input image and also the image intensities are normalised within a [0,255] region, as defined in [262].

6.3.6 Deep Feature Extraction

In this Chapter, 2 pretrained CNNs were used as deep feature extractors. The advantage of using a pretrained CNN to extract deep image features, as opposed to training a new CNN, are: (1) less computational power is needed as we are allowing the CNN to process each image only once to extract deep feature representations; (2) less data is required in order to achieve high accuracy results as layers deep in the CNN architecture contain activations that can be used for deep feature representations. CNNs have been trained to specifically determine and highlight key features in an image and pretrained CNNs allow images to be inserted and layers produce a response or activation to the image. These activations' or deep features as they will be called in this work, can be extracted in the form of a feature vector. The authors that created datasets Food-5K and Food-11 fine-tuned a GoogLeNet model, therefore for performance comparison, we adopted a different approach of using GoogLeNet, not for fine-tuning but for deep feature extraction and to use these deep features to train machine learning classifiers. As stated, the 2 CNNs we have chosen achieved high accuracy results when applied to ILSVRC ImageNet dataset [265]. The datasets used in this work are small in comparison to the datasets needed to train a CNN from scratch such as ILSVRC dataset which contains over 14 million images [265]. Figure 6.5 describes the pipeline used in this work where by images are processed to extract deep features to be used for classification.

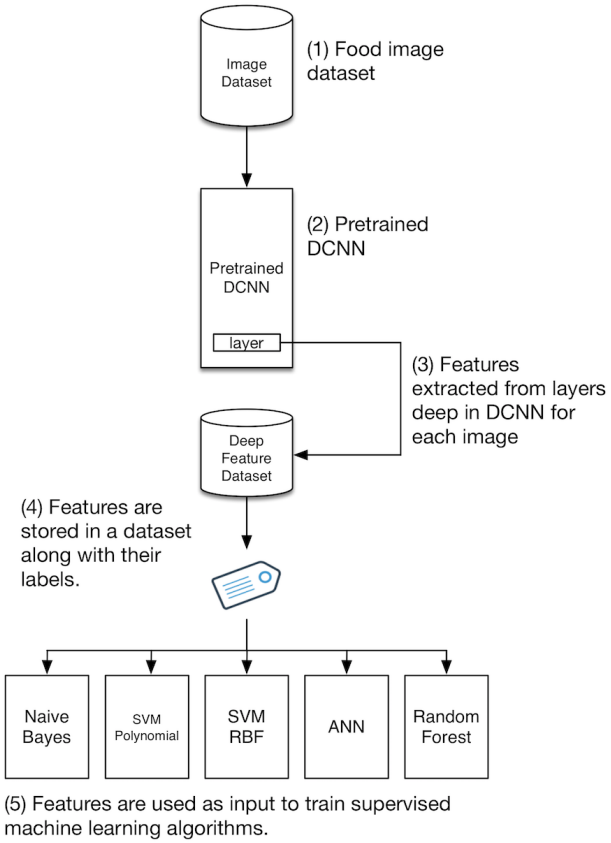


Fig. 6.5 Diagram describing the pipeline of deep feature extraction. (1) Food image datasets are used as input into (2) (pretrained CNN). (3)A layer deep in the architecture is specified and the image is processed by the CNN and the output (of the specified layer) is a generic image feature vector. (4) These generic image feature vectors can be collated to form a feature dataset and each feature vector generated by the CNN layer is labelled in accordance to the category from where the image taken from. (5) The generic image feature dataset can then be used as input to a range of conventional machine learning algorithm.

6.3.6.1 Layer Selection

To extract features from pretrained CNN, a layer needs to be selected for each model. During the training of CNN models, the output from convolutional layers and the pooling layers depict high level representations of images. In this study, we extracted deep feature maps immediately after the last pooling layer of each CNN to determine if these feature representations can accurately generalise between different food classes in food image dataset. The layer names used to extract deep features from CNN architecture are used to distinguish

between different layers in the pretrained CNN models. Table 6.3 lists the size of each pretrained CNN model and the chosen layer for deep feature extraction. Initial experiments were completed and results showed that features extracted from layer 'pool5' achieved higher percentage accuracy compared to features extracted from the last fully connected layer 'fc1000' for ResNet-152. In regards to GoogLeNet Inception model, layer'cls3_pool' was used instead of using the fully connected layer directly before softmax layer (as traditionally used for CNN deep feature extraction). Figure 6.6 and 6.7 are examples of residual activations extracted from a residual based CNN.

Table 6.3 Pretrained CNNs used as deep feature extractors in this work. This table lists the name of the CNN, the amount of layers present, the dataset used to train the CNN, and layer used in this work.

CNN	Layers	Trained	Layer
ResNet-152	152	ImageNet ILSVRC	pool5
GoogLeNet	22	ImageNet ILSVRC	cls3_pool

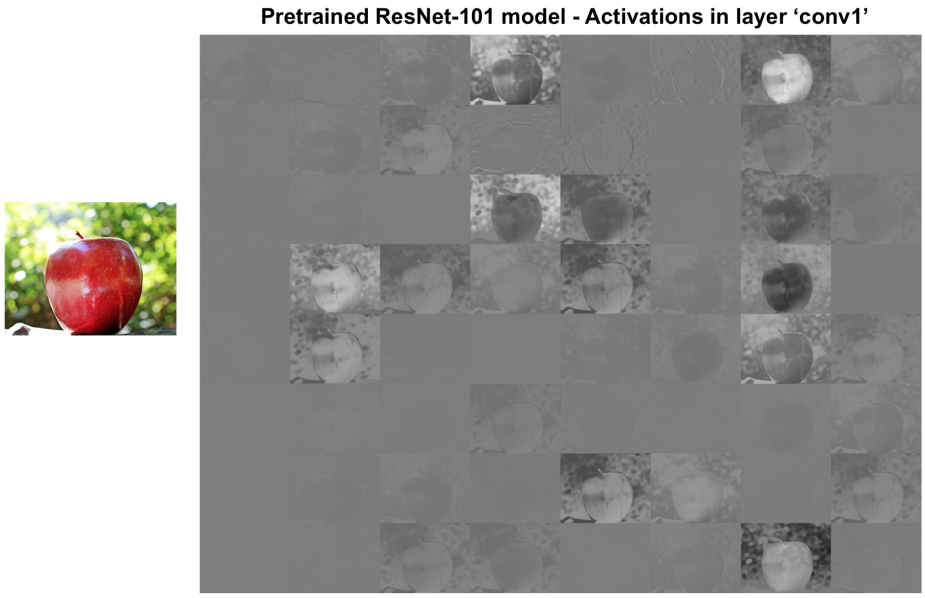


Fig. 6.6 Output of convolutional layer residual activations using a food image as input to a ResNet CNN. Each layer of a CNN consists of 2D arrays of channels and they are able to illustrate what areas or features of an image are 'activated'.

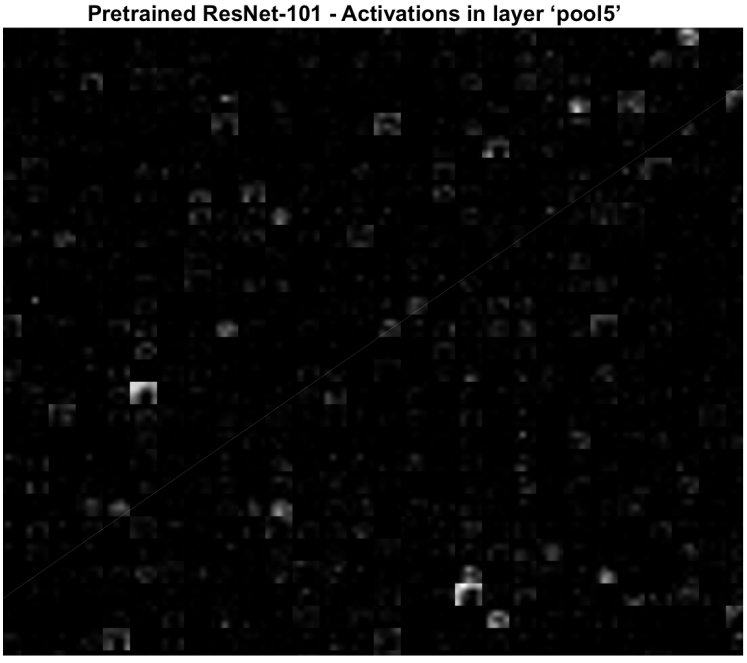


Fig. 6.7 Example of residual activations located deep in ResNet-101 convolutional layer 'pool5', the network learns to detect more complicated features. Deeper layers combine features from earlier layers to highlight detailed shape and features.

6.3.7 Pretrained Models using MatConvNet Package

MatConvNet is a Matlab library that allows for the training of state-of-the-art CNNs or to apply pretrained CNNs for deep feature extraction to be used for image classification [262]. In this work, MatConvNet was used to utilise 2 pretrained CNNs for deep feature extraction both trained on ILSVRC ImageNet dataset. MatConvNet packages allow for the fine-tuning of pretrained CNN [262]. In this work ResNet-152 and GoogLeNet were chosen to extract deep features to train classification models, the reason ResNet-152 was used was that it has achieved the lowest top-1 error of 23% using ILSVRC 2012 validation dataset in the MatConvNet package. GoogLeNet is another widely used CNN model available on MatConvNet package and was used for deep feature extraction in this work for performance comparison with the fine-tuned GoogLeNet model trained in [262].

6.3.8 ResNet-152 CNN

ResNet-152 is a deep residual pretrained CNN [206]. At the time of development, the authors of this CNN have described it as 'the deepest network ever presented on ImageNet' (2015) and is based on utilising 'extremely deep nets' with a depth of up to 152 layers. A residual learning framework which allows training of networks easier to converge and promote increased accuracy. The main advantages that residual networks contribute is the acceleration of speed in training networks, the effect of the vanishing gradient problem is reduced, and increasing the depth of the network which results in less parameters. ResNet-152 is made up of residual connections that allow important information to be transferred between layers. Residual connections allow a gradient to pass backwards directly through layers without losing vital information, in a regular CNN, the gradient must always pass through an activation layer. This can cause the gradient to diminish, to circumvent this problem, connections within a CNN are appended with a shortcut that allows gradients to pass through

thus decreasing the effects of vanishing gradient (information loss). Experiments using residual connects (ResNet-152) have reported increased accuracy and lower training times, in comparison to other state of the arts [206]. The authors of ResNet-152 compare their work with other established CNNs and state that this residual deep net is 8x deeper than VGG nets [206]. We used ResNet-152 pretrained CNN with the image datasets mentioned in this work for feature extraction. We selected 'pool5' layer deep in the ResNet-152 architecture and for each image an extracted a feature vector of 2048 was computed.

6.3.9 GoogLeNet - Inception

GoogLeNet was also used for deep feature extraction, and these features were combined with a variety supervised machine learning models. In [204] a deep convolutional network was proposed that can achieve state of the art classification and object detection accuracy by training the network using ImageNet dataset for Large Scale Visual Recognition Challenge 2014. The motivation for GoogLeNet was that larger CNNs might encounter the problem of overfitting as there is a large number of parameters used in the network. GoogLeNet's main contribution is the introduction of Inception modules that utilise the concept of using an approximation of sparse structure with repeated dense components. Dimensionality reduction is used to ensure computational complexity is kept to a minimum. Multiple convolutional filters are used with different sizes to ensure that there is sufficient coverage of information clusters. Before more computational expensive convolutions (3 x 3, 5 x 5) a convolutional after the previous layer for data reduction. The results of GoogLeNet incorporating these inception modules achieved 6.67% top-5 error percentage in classification performance in ILSVRC Classification Challenge 2014. In this work, we extracted the deep activations using the fully connected layer cls3_pool which has a 1024 vector dimension and is located after the last pooling layer in GoogLeNet [204].

6.3.10 Metrics for Performance Measurement

Several metrics were used to assess the performance of the trained models. The metrics that were selected to assess each model were percentage, recall, F1 score, Kappa, and Area Under the Receiver Operating Characteristic curve (AUC). The output of each model can be presented using a confusion matrix. A confusion matrix is a table that is able to summarise the prediction outcome of a model by classifying instances as positive (P) instances or negative (N) instances. Confusion matrix can further provide greater insight into prediction outcomes by classifying predicted instances as true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). Visually, the performance of a confusion matrix can be quickly assessed by inspecting the diagonal line of the confusion matrix, the stronger instances that are present in this diagonal line signifies better performance. The metrics used to assess the experiments can be derived from the confusion matrix such as recall (sensitivity), Ac, and F1 score. Recall can be described as metric that describes how many instances are classified correctly. The F1 score is a weighted average using precision and recall and is measured between 0 (worst) and 1 (best). For Food-5K the AUC values were also computed for each experiment due to being a binary classifier and Cohen's kappa was calculated for Food-11, RawFooT-DB, and Food-101. Cohen's kappa is a metric that is used to measure the inter-rater agreement between two label sets in a classification problem, we use Cohen's Kappa along with other metrics to describe experiment results.

6.3.11 Training, Validation, and Evaluation Data Partitions

To evaluate the performance of our trained models, validation and evaluation datasets were extracted and used from Food-5K, and Food-11. For RawFooT-DB, an evaluation dataset was used supplied by the authors [259]. For Food-5K, Food-11, and RawFooT-DB, the authors already partitioned the datasets into evaluation and validation sets (Table 6.4) and

in this work we used the same data splits to train and test our models. For Food-101, we split the data into 75:25 for training and testing. Authors of Food-101 provide training and testing splits with testing images cleaned of noise, however in this work we randomly shuffled images for training and testing partitions to test how ResNet-152 performs in classifying food images with noise and high food variance. This would give an indication of how deep features would perform in classifying difficult datasets such as Food-101. Table 6.4 is a summary of the data partitions used in this work for each food image dataset and the names for each partition follows the author's naming convention. Several metrics were computed during the experiment stage e.g. kappa statistic, F1 score, recall, average ROC, and accuracy to measure the performance of each trained model. Food-5K and Food-11 datasets each contained training, validation, and evaluation images. Training images were used for feature extraction to train machine learning classifiers. Validation images were used to determine if hyper-parameters used yield adequate results and evaluation dataset was to fully evaluate overall trained model. For RawFooT-DB, authors developed training and testing datasets by taking each image and dividing it into 16 tiles, 8 tiles are for training and the remaining 8 for testing. Each class contains 368 images (tiles) which represent 8 tile texture samples under 46 different lighting conditions. The testing dataset was used to verify if the trained model able to generalise between food texture classes. Food-101 dataset was randomly partitioned; 75% for training and 25% for testing. Testing partition was used to verify trained Food-101 classifiers. UNICT-FD889 and Caltech-101 testing datasets were used to further evaluate trained food/non-food classification models.

Table 6.4 Evaluation and testing methods used for each food image dataset. * denotes dataset splits supplied by dataset authors.

Dataset	Dataset Partition
Food-5K	Training, validation & evaluation*
Food-11	Training, validation & evaluation*
RawFooT-DB	Training & testing*
Food-101	75:25 training & testing
UNICT-FD889	Testing
Caltech-101	Testing

6.3.12 Weka Platform

In order to train the machine learning algorithms, Weka 3.8.1 [244] platform was used. Weka is a software application that contains various machine learning algorithms written in Java and the application was developed at University of Waikato, New Zealand. The application can be used for different tasks such as clustering, classification, visualisation, feature selection, and preprocessing and is very popular within universities for its ease of use. It is also popular because of the amount of algorithms available. The main reason that Weka 3.8.1 was used in this work was the detailed evaluation results output computed, which are collated into a window after evaluation has finished. Another major advantage of using Weka is the evaluation process in that a range of detailed metrics are computed for each class to describe the performance of the model. A confusion matrix is also computed to determine the performance of individual classes for the trained model using K-fold class validation or a dedicated validation dataset. The amount of machine learning algorithms that are available is also a factor in using Weka as well the easy to use graphical user interface (GUI). In this

work, Weka 3.8.1 will used with the extracted features from image datasets for classification, analysis, and evaluation [244].

6.3.12.1 WekaPython Plugin & Scikit-Learn

WekaPython plugin was used with Weka 3.8.1 that allows the training of scikit-learn [266] machine learning classifiers. The wekaPython package relies on Python version 2.7 or higher being installed on the user's system and uses a range of Python packages to function correctly such as pandas, numpy, scikit-learn, and matplotlib. In this work, the wekaPython was used to train and evaluate the deep features extracted from the pretrained CNNs. Weka was used to train an ANN for experiments with Food-101. Due to its flexibility for working with larger datasets, Python v2.7.10 with scikit-learn library was also used to train the other machine learning classifiers for the Food-101 dataset. The following machine learning algorithms were used in this work [266]:

6.3.12.2 Machine Learning Models Used

Naive Bayes is a popular machine learning algorithm known for their efficiency and minimal processing. They can be described as a set of simple probabilistic classifiers derived from Bayes' Theorem. The term 'naive' is used to describe the algorithm because it assumes that attributes are independent of the associated class. Bayes rule is enforced to compute the probability of a class based upon the values in the vector. Bayes' rule of conditional probability states that if you have a hypothesis H and the evidence (feature attributes) is connected to that hypothesis [134]. Naive Bayes assumes independence and the algorithm works efficiently and can outperform the most sophisticated machine learning algorithms on certain datasets. In this work, a Gaussian naive bayes classifier was trained using the extracted CNN deep features. A Gaussian naive bayes classifier is used when continuous values are present by assuming a normal distribution in the dataset as the mean and standard

deviation is computed for each class. SVMs are able to implement the use of non-linear boundaries by using kernels (e.g. RBF, Polynomial) to transform feature representation into a higher dimensional space to predict multiple classes. In classification problems, the use of SVM have performed well in generalising on a variety of classification problems such as food classification, face detection, and object detection. In some problems the training data in a problem may become inseparable meaning that there is not a clear boundary definition, SVMs are able to enforce nonlinear boundaries in transformed feature spaces [267]. In this work, we train 2 C-SVM models using Polynomial kernel and Radial Basis Function (RBF). C-SVM uses a C regularisation parameter that implements a weight penalty for misclassifications to improve the accuracy of the model. Also in this work, ANNs were trained for each dataset using a Weka plug-in [131] with the following parameters listed in Table 6.5. The learning rate was set to adaptive unless otherwise stated in the experiments. The adaptive learning rate function uses a number of base learning rates on the training data to determine the most suitable by comparing the cost function of each. The Weka plugin uses dropout regularisation to prevent over fitting and Rectified Linear Units (ReLU) as the activation functions [131, 268]. Scikit-learn was used to implement Random Forest model and Table 6.6 states the parameters used for RF models used in this work.

1. Gaussian Naive Bayes (wekaPython scikit-learn)
2. Support Vector Machines (SVM) (wekaPython scikit-learn)
3. Artificial Neural Network (ANN)
4. Random Forest Classifier (wekaPython scikit-learn)

For Food-101 food image dataset, datasets were manually split 75:25 and the follow parameters were used to split and shuffle the dataset to train and test each machine learning classifier;

1. Gaussian Naive Bayes - random_state 1
2. Support Vector Machines - random_state 1
3. Artificial Neural Network - random_seed 1
4. Random Forest Classifier - random_state 1

Table 6.5 Hyper-parameters used for each ANN.

ANN	**Parameters**
Number of iterations	1000 (max)
Num of layers	1
Neurons per layer	100
Learning rate	Adaptive*
Learning momentum	0.2
Weight penalty	0.00000001 (default)
Hidden Layers drop out rate	0.5
Input layer drop out rate	0.2
Activation function	ReLu
Convergence threshold	0.2
Batch	100

6.3.12.3 Random Forest

In this work a scikit-learn Random Forest classifier was used with wekaPython and table 8 lists the parameters used for this model. Table 6.6 states the parameters used for RF models used in this work

Table 6.6 Table showing hyper-parameters used for Weka Random Forest classifier. Hyper-parameters used for this classifer are default.

Random Forest	Parameters
Criterion	entropy
Number of estimators	50
Random State	none
Depth of tree	None
Minimum number of samples split	2
Minimum number of samples for leaf node	1
Number of features for best split	auto
Bootstrap	True
Max leaf nodes	None
Random State Instance	None
Max depth	None
Minimum num of leaf samples	1

6.4 Experimental Results

6.4.1 Food /Non-Food Classification Results

6.4.1.1 Food-5K

This section lists the results of our experiments using the food image datasets. Tables 6.7 and 6.9 list the detailed results of Food-5K. Accuracy, recall, F1 score, and ROC values were used to measure the performance of each the classification models for both validation and evaluation datasets. Initial results show that deep features combined with machine

learning classifiers achieved high accuracy results when distinguishing between food and non-food images. The use of SVM with RBF kernel achieved the highest accuracy with 99.4% using ResNet-152 for deep feature extraction with validation dataset and 98.8% with evaluation dataset. Table 6.7 and 6.9 also lists the confusion matrices of using SVM-RBF with ResNet-152 to detect food images in validation dataset and ANN with ResNet-152 features to detect food images in evaluation dataset. GoogLeNet deep features achieved marginally lower accuracy results, however for the evaluation dataset, GoogLeNet deep features with ANN achieved the same accuracy result as SVM-RBF and Random Forests classifier with ResNet-152 features with 98.8%. In regards to using SVM classifiers in Food-5K, the use of the RBF kernel achieved marginally higher accuracies compared to the polynomial kernel and Gaussian naive bayes achieving the lowest accuracy results in both testing datasets with both deep feature types.

Table 6.7 Classification results using ResNet-152 and GoogLeNet to extract deep activations (extracted from Food-5K) with supervised learning algorithms. * denotes highest accuracy achieved.

Food-5K - Validation								
Model	**ResNet-152 - pool5**				**GoogLeNet - cls3_pool**			
	Acc (%)	**Recall**	**F1**	**ROC**	**Acc (%)**	**Recall**	**F1**	**ROC**
NB	98.7	0.99	0.99	0.99	97.5	0.98	0.98	0.99
SVM (RBF)	99.4*	0.99	0.99	0.99	98.5	0.99	0.99	0.99
SVM (Poly)	99	0.99	0.99	0.99	98.5	0.99	0.99	0.99
ANN	99.2	0.99	0.99	1	99	0.99	0.99	0.99
RF	98.9	0.99	0.99	1	98.6	0.99	0.99	0.99

Table 6.8 Confusion matrix showing results of highest accuracy results achieved using ResNet-152 features classifying validation dataset of Food-5K using a SVM with RBF kernel.

Confusion Matrix using SVM-RBF with ResNet-152 Validation Dataset Features

		Predicted Labels	
		Food	Non-Food
True	Food	498	2
	Non-Food	4	496

Table 6.9 Classification results using ResNet-152 and GoogLeNet to extract deep activations (extracted from Food-5K) with supervised learning classifiers using evaluation dataset. * denotes highest accuracy achieved.

Food-5K - Evaluation								
Model	**ResNet-152 - pool5**				**GoogLeNet - cls3_pool**			
	Acc (%)	**Recall**	**F1**	**ROC**	**Acc (%)**	**Recall**	**F1**	**ROC**
NB	97.3	0.97	0.97	0.98	96	0.96	0.96	0.98
SVM (RBF)	98.8*	0.99	0.99	0.99	98.3	0.98	0.98	0.98
SVM (Poly)	98.3	0.98	0.98	0.98	98.2	0.98	0.98	0.99
ANN	98.8*	0.99	0.99	0.99	98.8*	0.99	0.99	0.99
RF	98.8*	0.99	0.99	0.99	98.5	0.99	0.99	0.99

Table 6.10 Confusion matrix showing results of highest accuracy results achieved using ResNet-152 features classifying evaluation dataset of Food-5K using ANN.

Confusion Matrix using ANN with ResNet-152 Evaluation Dataset Features

		Predicted Labels	
		Food	Non-Food
True	Food	493	7
	Non-Food	5	495

To further test our models, experiments were conducted that tested food/non-food trained models on the Food-11 dataset as what was completed in [13] for more detailed comparison. Food-11 dataset contains 16,643 images and they are all classed as food images, GoogLeNet and ResNet-152 deep features were used to extract deep features from Food-11

and used with SVM-RBF and ANN models to classify them to detect food in the images. Table 6.11 is a breakdown of the results using our methods to classify Food-11 dataset.

6.4.1.2 UNICT-FD889 & Caltech

Table 5.11 list the results of using SVM-RBF and ANN trained with Food-5K training ResNet-152 deep features for classifying UNICT-Caltech, which combines images in UNICT-FD889 and Caltech-101 to make a food/non-food dataset. UNICT-Caltech dataset is a larger dataset and using this dataset with our trained models allows us to get a better indication how ResNet-152 features perform in detecting food in images.

Table 6.11 Results comparison of classifying Food-11 and UNICT-Caltech with our Food/Non-Food classification models. * denotes highest percentage accuracy achieved for both datasets. * denotes highest accuracy achieved.

Method	**Num of Food Images Detected**	**Accuracy**
ResNet-152 + ANN (Food-11)	16, 208	97.39%*
ResNet-152 + SVM-RBF (Food-11)	16,176	97.19%
GoogLeNet + ANN (Food-11)	16,171	97.16%
GoogLeNet + SVM-RBF (Food-11)	15,646	94.00%
ResNet-152+ SVM-RBF (UNICT-Caltech)	12,409	97.50%*
ResNet-152+ ANN (UNICT-Caltech)	12,283	96.51%

6.4.2 Food Item Classification Results

6.4.2.1 Food-11

Results show that using ResNet-152 and GoogLeNet deep features are able to achieve high accuracies when classifying across major food groups. Results are presented in Tables 6.12 and 6.13. The maximum accuracy achieved was using ANN for both ResNet-152 and GoogLeNet features achieving 91.34% and 86.44% respectively with evaluation dataset. For ResNet-152 features an F-measure of 0.91 was achieved and 0.86 with GoogLeNet features using ANN. For the ANN trained using ResNet-152 features, the base learning rate was set to auto-detect which allows the ANN Weka plugin to initially test various learning rates to determine the lowest cost function. Initial tests revealed that 1.0 learning rate achieved the lowest cost function and the ANN used that to learning rate to initially begin the training. The learning rate decreased over the course of the training if the network cost function didn't improve after 10 mini-batch iterations. The network converged after 204 iterations ending with a learning rate of 0.01. Further analysis revealed the SVM models trained with RBF and Polynomial kernel using ResNet-152 features achieved 89.99% and 88.86% accuracy respectively and 85.36% and 86.05% using GoogLeNet features using evaluation dataset. Figure 6.8 shows the confusion matrix of using an ANN trained with ResNet-152 features to classify the evaluation dataset. Figure 6.9 is an example of different types of food categories that were misclassified as shown in the confusion matrix in Figure 6.8. Table 6.14 also lists the Top-4 misclassifications for each group.

Table 6.12 Classification results using ResNet-152 and GoogLeNet to extract deep features (extracted from Food-11) with supervised learning classifiers. * denotes highest accuracy achieved.

Food-11 - Validation Dataset								
Model	**ResNet-152 - pool5**				**GoogLeNet - cls3_pool**			
	Acc (%)	**Recall**	**F1**	**Kappa**	**Acc (%)**	**Recall**	**F1**	**Kappa**
GNB	73.03	0.73	0.73	0.70	67.49	0.68	0.68	0.64
SVM (RBF)	88.11	0.88	0.88	0.87	82.36	0.82	0.82	0.80
SVM (Poly)	86.65	0.87	0.87	0.85	83.70	0.84	0.84	0.82
ANN	89.18*	0.89	0.89	0.88	84.11	0.84	0.84	0.82
RF	78.43	0.78	0.78	0.76	75.48	0.76	0.75	0.72

Table 6.13 Classification results using ResNet-152 and GoogLeNet to extract deep features (extracted from Food-11) with supervised learning algorithms.

Food-11 - Evaluation Dataset								
Model	**ResNet-152 - pool5**				**GoogLeNet - cls3_pool**			
	Acc (%)	**Recall**	**F1**	**Kappa**	**Acc (%)**	**Recall**	**F1**	**Kappa**
GNB	75.38	0.75	0.76	0.72	69.73	0.70	0.70	0.66
SVM (RBF)	89.99	0.90	0.90	0.89	85.36	0.85	0.85	0.84
SVM (Poly)	88.86	0.89	0.89	0.87	86.05	0.86	0.86	0.84
ANN	**91.34**	0.91	0.91	0.90	86.44	0.86	0.86	0.85
RF	80.40	0.80	0.80	0.78	78.24	0.78	0.78	0.75

Results Comparison of classifying Food-11 using ANN trained with ResNet-152 features.

Classified as:

bread	dairy	dessert	egg	fried	fruit/veg	meats	pasta	rice	seafood	soup	
324	2	7	11	9	2	8	0	1	2	2	bread
0	121	17	3	1	0	1	0	1	3	1	dairy
9	9	430	17	3	2	13	0	1	5	11	dessert
21	2	9	293	0	1	5	0	0	3	1	egg
5	1	5	6	255	0	7	0	2	2	4	fried
0	1	3	1	0	225	0	0	0	1	0	fruit/veg
4	1	8	5	7	0	401	1	1	3	1	meats
0	0	0	0	0	0	0	147	0	0	0	pasta
0	0	1	0	0	0	1	0	93	0	1	rice
4	2	5	4	1	1	3	0	1	281	1	seafood
1	0	5	1	0	0	0	1	0	5	487	soup

Fig. 6.8 Confusion matrix of Food-11 classes using ANN model trained using ResNet-152 features.

Fig. 6.9 Example of Food-11 classes which are misclassified based on confusion matrix generated from ANN model trained using ResNet-152 features. Images highlight shared characterisitics that could lead to misclassifications.

Table 6.14 Misclassifications between food groups.

Food Group	Top-4 Group Misclassifications			
	Group 1	Group 2	Group 3	Group 4
Bread	Egg	Fried	Meats	Dessert
Dairy	Dessert	Egg	Seafood	-
Dessert	Egg	Meats	Soup	Dairy
Egg	Bread	Meats	Seafood	Dairy
Fried	Meats	Egg	Dessert	Bread
Fruit & Veg	Dessert	-	-	-
Meats	Dessert	Fried	Egg	Bread
Pasta	-	-	-	-
Rice	Soup	Meats	Dessert	-
Seafood	Dessert	Egg	Bread	Dairy
Soup	Seafood	Dessert	Bread	Egg

6.4.2.2 RawFooT-DB

Results listed in Table 6.15 reveal ResNet-152 features trained with SVM and RBF kernel achieved an accuracy of 99.10% and our ANN also with ResNet-152 99.28% in classifying RawFooT-DB. The results show that deep features efficiently classify isolated texture images with an accuracy of 99.28% across various lighting conditions and further investigation analysing the confusion matrix generated from SVM-RBF model shows that there were a number of classes that experienced misclassifications. For example, several instances were wrongly classified as chickpeas instead of white peas. Investigating the images from both categories, it was clear that there are similarities between shape, colour, and texture as shown in Figure 6.10. When investigating the ANN confusion matrix, several white pea instances were also classed as chickpeas and there were also several mango instances classed as apple slices (Figure 6.10). Figure 6.12 is also an example of image classes that were misclassified using an ANN (chicken breast and milk chocolate) and there are clear similarities between each class. These images showed similar characteristics in colour and texture, similarly hamburger images were classified as salami and further investigation showed very similar texture, colour, and patterns however ResNet-152 features still achieved 0.98 F-measure for hamburgers and 0.99 for salami and an overall F-measure of 0.99.

Table 6.15 Classification results using ResNet-152 and GoogLeNet to extract deep features (extracted from RawFoot dataset) with supervised learning classifiers. * denotes highest accuracy achieved.

RawFoot Dataset - Training/Testing Split								
Model	**ResNet-152 - pool5**				**GoogLeNet - cls3_pool**			
	Acc (%)	**Recall**	**F1**	**Kappa**	**Acc (%)**	**Recall**	**F1**	**Kappa**
GNB	82.02	0.82	0.83	0.82	78.42	0.78	0.79	0.78
SVM-RBF	99.10	0.99	0.99	0.99	96.63	0.97	0.97	0.97
SVM-Poly	98.21	0.98	0.98	0.98	96.74	0.97	0.97	0.97
ANN	99.28*	0.99	0.99	0.99	97.04	0.97	0.97	0.97
RF	98.13	0.98	0.98	0.98	94.03	0.94	0.94	0.94

Fig. 6.10 Example of RawFooT-DB classes which are misclassified based on confusion matrix generated from SVM-RBF model trained using ResNet-152 features. Images highlight shared characterisitics that could lead to misclassifications.

.

For further analysis using RawFooT-DB with ResNet-152 and GoogLeNet features, we reordered the food types into 7 groups, vegetables, rice/grains/wheat/seeds, fruits, sweets, breads, meat/fish, and miscellaneous (e.g. coffee, powders, sugar). Figure 6.13, 6.14, and 6.15 show the F-measure of the food texture types rearranged into food groups for ANN and SVM-RBF models. It is clear the from Figure 6.13, 6.14, and 6.15 that there is a decrease in accuracy in 'meat/ fish' group. Further investigation show that many of the foods in meat category share similar characteristics and therefore many meat based classes were misclassified as other meat classes as shown in Figure 6.11.

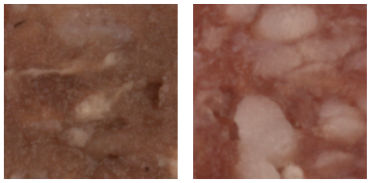


Fig. 6.11 Example of RawFooT-DB classes which are misclassified based on confusion matrix generated from ANN model trained using ResNet-152 features (**hamburger and salami**).

Other unrelated food types also share similar texture properties e.g. chicken breast shares similar characteristics with other textures such as 'milk chocolate' (Figure 6.12). Figure 6.13, 6.14, 6.15 also show decrease in accuracy with chickpeas and white peas due to sharing texture and shape characteristics and this is evident in Figure 6.15 using GoogLeNet deep features with ANN. Figure 6.16, 6.17, and 6.18 lists reordered F-measure of food classes in RawFooT-DB for models that achieved highest accuracies.

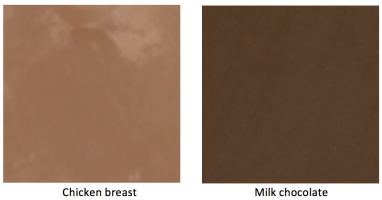


Fig. 6.12 Example of RawFooT-DB classes which are misclassified based on confusion matrix generated from ANN model trained using ResNet-152 features (**chicken breast and milk chocolate**)

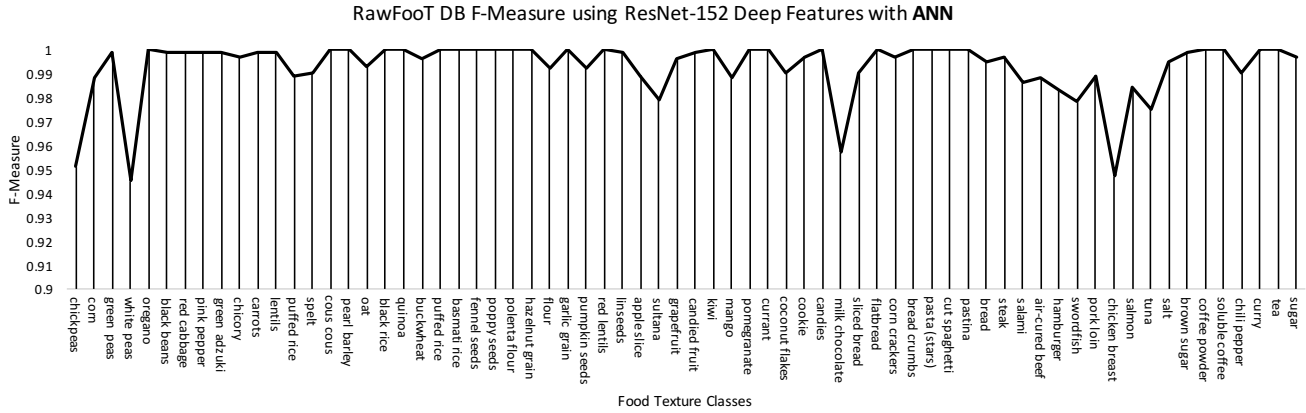


Fig. 6.13 RawFooT-DB F-Measure of reordered classes by major food groups using ResNet-152 features with ANN.

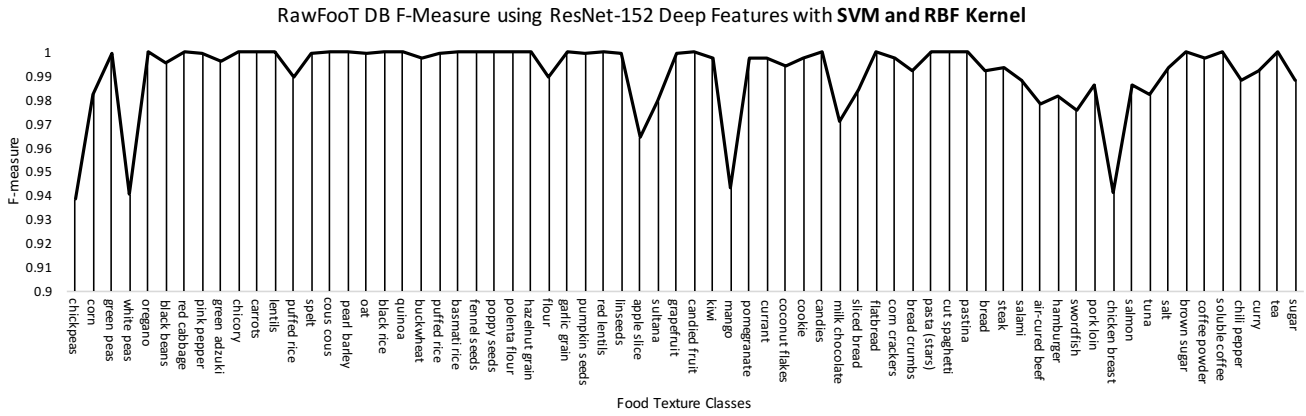


Fig. 6.14 RawFooT-DB F-Measure of reordered classes by major food groups using ResNet-152 features with SVM with RBF kernel.

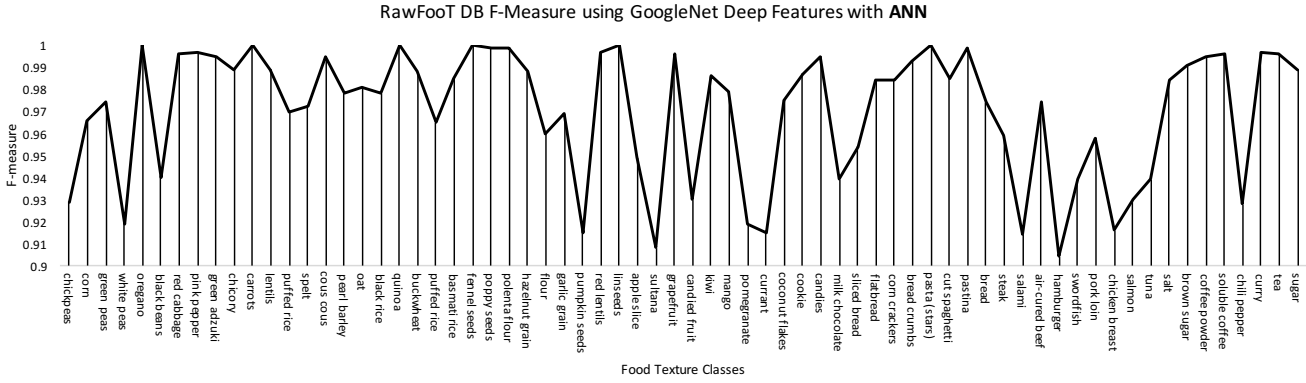


Fig. 6.15 RawFooT-DB F-Measure of reordered classes by major food groups using GoogLeNet features with ANN.

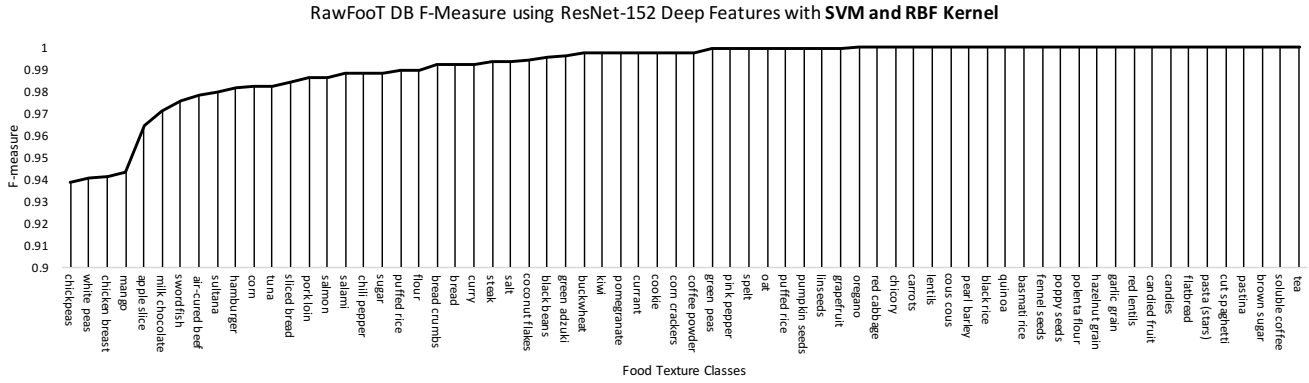


Fig. 6.16 RawFooT-DB F-Measure of reordered classes by major food groups using ResNet-152 features with SVM-RBF.

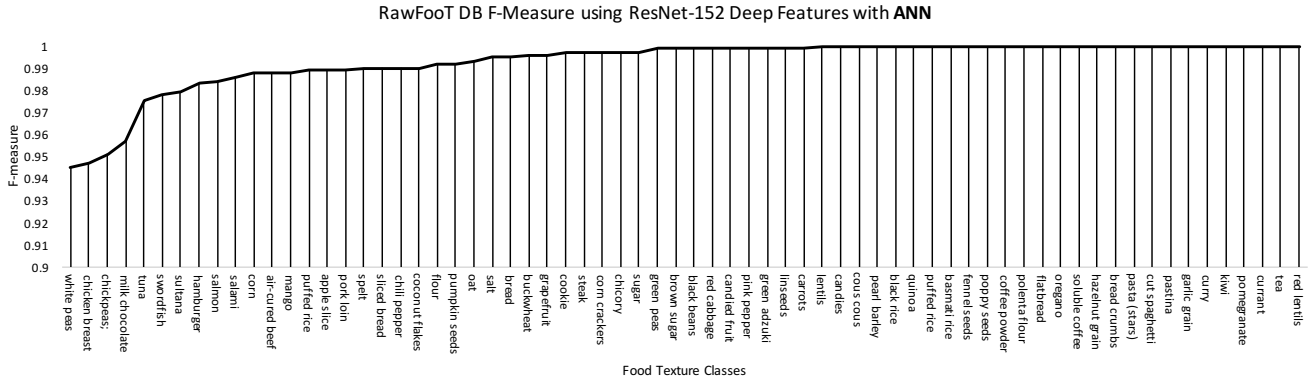


Fig. 6.17 RawFooT-DB F-Measure of reordered classes by major food groups using ResNet-152 features with ANN.

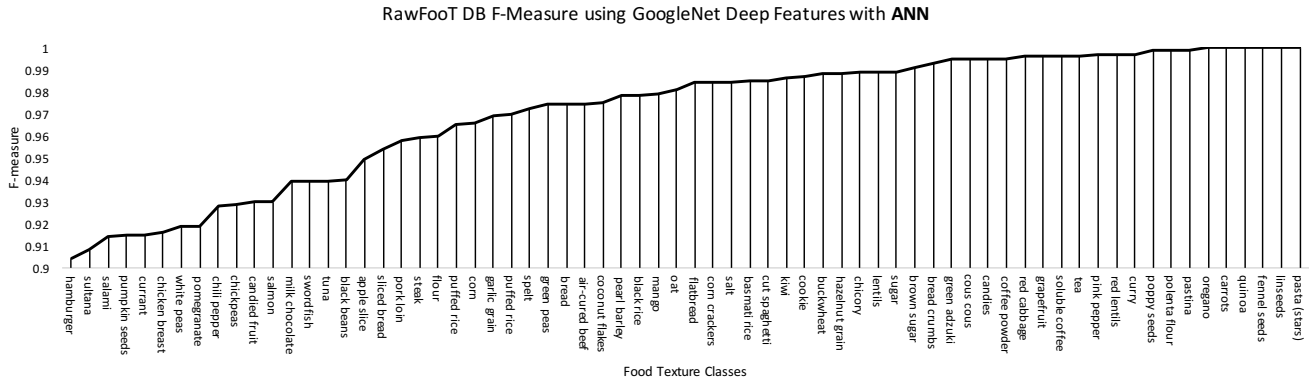


Fig. 6.18 RawFooT-DB F-Measure of reordered classes by major food groups using GoogLeNet features with ANN.

6.4.2.3 Food-101

From previous experiments using Food-5K and Food-11, and RawFooT-DB, ResNet-152 deep features achieved the highest accuracies. ResNet-152 deep features was used for classifying Food-101, which can be described as fine-grained food image dataset that contains similar food items (i.e. different kind of soups, meats images taken in a free living environment). Results listed in Table 6.16 show that ANN and SVM-RBF along with ResNet-152 features achieved the highest accuracy across for Food-101 achieving 64.98%. To train the ANN, Food-101 was partitioned into 75:25, training and testing, with random seed of '1' using Weka 3.8.1 (same ANN plug-in used with other experiments for Food-5K, Food-11, and RawFooT-DB). To train the ANN, the learning rate was initially set to 1 with mini-batch gradient descent. For the other classification models Python 2.7.10 with Scikit v0.19 was used instead of Weka because Python v2.7.10 and scikit-learn offers more flexibility and other libraries could be used, as well as its ease of use when working with larger datasets and for data analysis. The parameters for the classifiers remained the same as other experiments with Weka as wekaPython contains the same models as scikit-learn. To train the other classifiers using scikit-learn, Food-101 was also split in 75:25 training and testing with a random state parameter of '1'. Table 6.16 shows the accuracy, recall, F-Measure, and kappa statistic of using ResNet-152 deep features. The results are much lower than previous experiments with the highest accuracy with 64.18% for ANN and 64.98% for SVM-RBF. This could be based on a number of factors; (1) Food-101 is a much larger dataset with 101 food classes with other 1,000 images in each class, (2) Images in Food-101 were captured in free-living environments under different lighting intensities and conditions. Noise is also highly prevalent in Food-101 with non-food items and other unrelated food items present in images. (3) Food-101 also contains 'fine-grained' classes as some classes may contain similar characteristics to other classes e.g. different types of soup dishes or rice based dishes. These type of image classes are challenging and coupled with other factors such as lighting conditions, could result in

misclassifications. The kappa statistic was also generated for ANN and SVM-RBF at 0.64 and 0.65 respectively, which indicates statistical agreement.

Table 6.16 Classification results using ResNet-152 to extract deep activations (extracted from Food-101 dataset) with supervised learning algorithms. Highest accuracy denoted by *.

Food-101 Dataset - 75:25 training/evaluation				
Model	**ResNet-152 - pool5**			
	Acc	**Recall**	**F1**	**Kappa**
GNB	45.64%	0.46	0.46	0.45
SVM-RBF	64.98%*	0.65	0.65	0.65
SVM-Poly	63.04%	0.63	0.63	0.63
ANN	64.18%	0.64	0.64	0.64
RF	39.33%	0.39	0.38	0.39

Some misclassifications occurred across different classes in Food-101 experiments. Figure 6.19 and 6.20 is an example of typical food classes that were misclassified. Misclassifications occured with the steak food class with both the ANN and SVM-RBF. Steak instances were wrongly classified as pork chop, prime rib, and filet mignon using SVM-RBF and ANN, similarly, several pork chop instances were classified as steak, prime rib, and foie gras. This may be due to the shared characteristics of shape, texture, and colour. In regards to the desserts, several items were wrongly classified, the panna cotta class was wrongly classified as a cheese cake, and chocolate mousse and the cheese cake class was wrongly classified as a panna cotta, chocolate mousse, chocolate cake, and strawberry shortbread. Further investigation showed that these classes share similar characteristics such as shape and colour which may contribute to them being wrongly classified. Beignets were also wrongly

classified as donuts. Investigation showed that beignets are very similar to donuts regarding appearance, texture, colour, and shape, however, SVM-RBF trained with ResNet-152 features were still able to achieve an F-measure of 0.77 for beignets.

Figure 6.21 shows the F-measure for each food class in Food-101 for SVM. For further analysis, we organised the food classes into groups. Images were allocated into groups; (1) breads, pasta, (2) desserts, (3) eggs, (4) fried foods, (5) meats and fish, (6) mixed foods (foods that contained a mixture of foods) and (7) vegetables. Foods were organised into different foods to determine if ResNet-152 features had any inherent advantage for classifying specific food groups. The average F-measure was computed for each group, and the vegetable group achieved the highest with an average F-measure of 0.71 using SVM-RBF model. However, it should be noted that the vegetable category contained a small number of images in comparison to other groups. In regard to using SVM-RBF model to classify specific food items, the class the achieved the highest F-measure was ‘edamame’ with 0.98, and further investigation showed that ‘edamame’ images are very similar as the food item is distinct and there is little variation with the ‘edamame’ food type, and also they are the same shape and colour. The food item that achieved the lowest F-measure was ’steak’ with an F-measure of 0.36. Steak food class experienced misclassifications with other food types with other meat classes e.g. pork chop, prime rib, and foie gras due to the similar shape, colour, and texture (Figure 6.19). In regards to using ANN model, ‘edamame’ also achieved the highest with 0.97 F-measure and ’steak’ was also the lowest with 0.30. Figure 6.22 lists the lowest to highest F-measure for each food class in Food-101 to determine what food classes performed the worst and best using ResNet-152 features with SVM-RBF model.

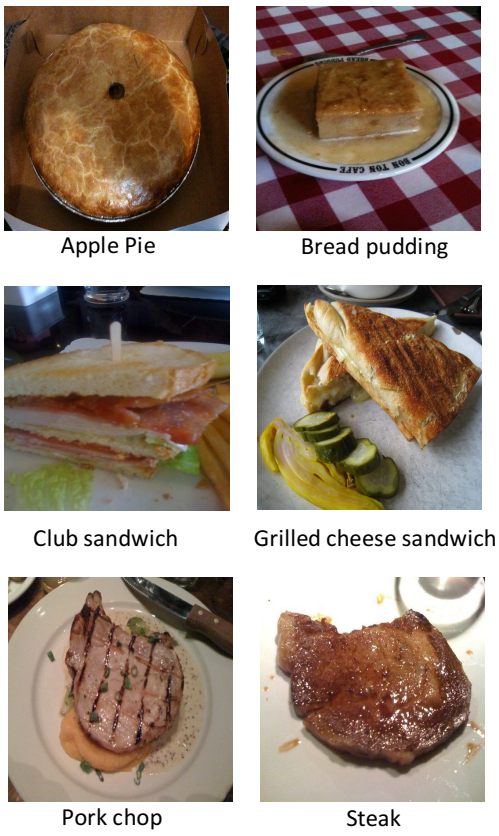


Fig. 6.19 Example of Food-101 classes which were misclassified based on confusion matrix generated from ANN and SVM-RBF models trained using ResNet-152 features. Food classes are on the left experience misclassification with the food classes on the right.

.

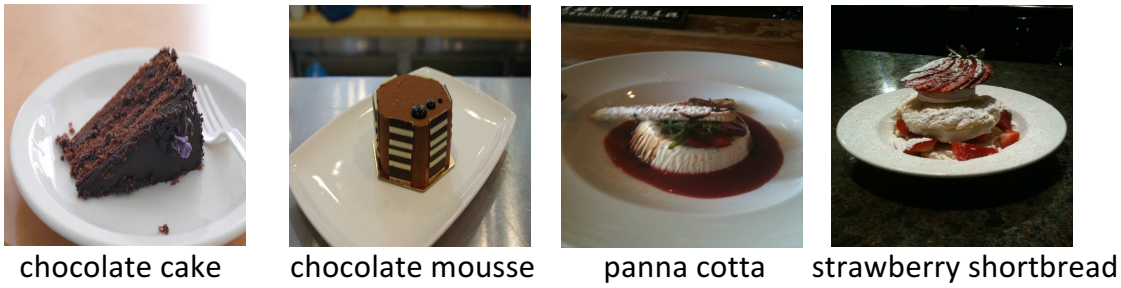


Fig. 6.20 Example of Food-101 dessert classes which were misclassified based on confusion matrix generated using both SVM-RBF and ANN models trained with ResNet-152 features.

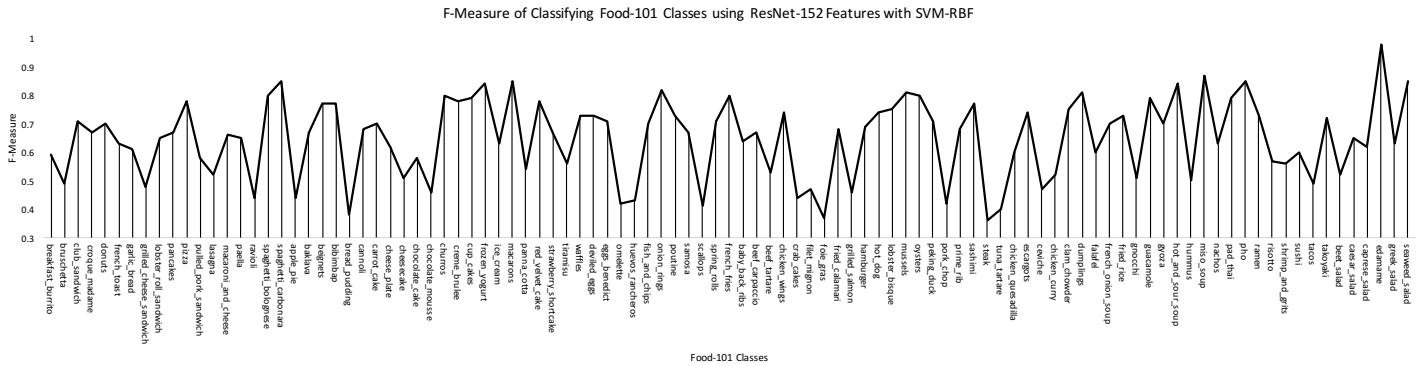


Fig. 6.21 Food-101 F-Measure of reordered classes by major food groups using ResNet-152 features with SVM with RBF kernel.

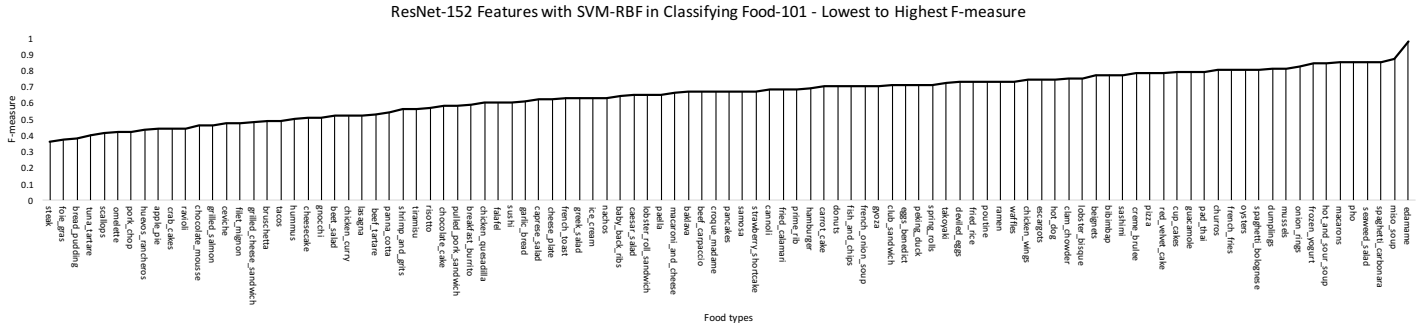


Fig. 6.22 Food-101 F-Measure of reordered food classes by lowest to highest F-measure using ResNet-152 features with SVM with RBF kernel.

6.5 Discussion

In this work, we used deep features extracted from pretrained CNNs for food image classification. We compared 2 popular pretrained CNNs, ResNet-152 and GoogLeNet and extracted deep features from layers deep in each CNN architecture to classify Food-5K, Food-11, and RawFooT-DB. For Food-101 ResNet-152 deep features were selected as it consistently achieved higher accuracies across other image datasets. We extracted a deep feature vector immediately after the last pooling layer in each architecture for each pretrained CNN for each from various food image datasets. From these experiments, we found that ResNet-152 achieved consistently higher results in Food-5K, Food-11, and RawFooT-DB

and because of this ResNet-152 features were used with Food-101. Food-101 is a much more difficult dataset due to the number of clsses and variation in images. Many classes contain low in-between class variance as many dishes are similar as shown in Figure 6.19 and 6.20. From the experiments, it was clear that using ResNet-152 can achieve high accuracies for Food-5K, Food-11 dataset, RawFooT-DB, and moderate accuracy for Food-101.

In regards to Food-5K, the deep features were able to detect food in images with high accuracy across all machine learning classifiers, achieving over 90% accuracy in each experiment. Trained food detection models were further benchmarked using against results obtained by the authors of Food-5K and Food-11 datasets who used a fine-tuned GoogLeNet [209] and the results presented in this Chapter suggest that there is potential to achieve high accuracies and performance without the need of fine-tuning pretrained CNNs for certain datasets and problems. Furthermore, due to the nature of Food-5K being a binary decision between food and non-food classes, generic deep features may be sufficient enough to provide adequate generalisation to classify between two classes (i.e. food and non-food).

ANN and SVM-RBF trained with ResNet-152 features achieved the highest accuracies in the majority of Food-5K experiments and the Food-5K ANN, and SVM-RBF model was further evaluated by classifying the entire Food-11 dataset for food detection. Results show that our ANN model trained using ResNet-152 features achieved higher food detection accuracy compared to the fine-tuned GoogLeNet model in [209] when tested against Food-11 image dataset as stated in Table 6.17. We also evaluated both our Food/Non-Food SVM-RBF model trained with ResNet-152 and GoogLeNet deep features using Food-11 for food detection and results showed that these models achieve marginally higher results compared to other results obtained in also listed in Table 6.17 [209].

Authors in [13] achieved 83.6% with Food-11 evaluation dataset and in our work ResNet-152 features with ANN achieved 91.34% and 89.99% with SVM-RBF, this is an improvement of 7.74% and 6.39% respectively. For Food-5K, ResNet-152 features achieved 98.8% in classifying Food-5K evaluation dataset and authors in [209] achieved 99.2%. Authors in [209] evaluated their food detection model using all images in the Food-11 dataset. This approach was performed in this Chapter and Table 6.17 compares our results. ANN and SVM trained with ResNet-152 deep features achieved marginally higher results than obtained in [209] with 97.39% and 97.19% respectively. GoogLeNet deep features with ANN also produced slightly higher results with 97.16% compared to proposed Fine-tuned GoogLeNet method in [209].

Fig. 6.23 Food image classes from Food-101 that share similar characteristics. Categories from left to right; french onion soup, hot and sour soup, clam chowder, miso soup

Table 6.17 Method and results comparison using Food-5K and Food-11. Bold font denotes accuracy improvement.

Author	**Method**	**Accuracy**	**Food Dataset**
Singla, et al. [209]	GoogLeNet (fine-tuned)	99.2%	Food-5K
Singla, et al. [209]	GoogLeNet (fine-tuned)	83.6%	Food-11
This work	ResNet-152 + ANN	98.8%	Food-5K
-	ResNet-152 + ANN	**91.34%**	Food-11
-	ResNet-152 + SVM-RBF	**89.99%**	Food-11
-	ResNet-152 + SVM-Poly	**88.86%**	Food-11

Table 6.18 Results comparison of classifying Food-11 with our Food/Non-Food classification models. Bold font denotes accuracy improvement.

Method	Num of Food Images Detected	Accuracy
Fine-Tuned GoogLeNet [209]	16,127	96.9%
ResNet-152 + ANN	16, 208	**97.39%**
ResNet-152 + SVM-RBF	16,176	**97.19%**
GoogLeNet + ANN	16,171	**97.16%**
GoogLeNet + SVM-RBF	15,646	94.00%

Table 6.18 also shows GoogLeNet features used to detect food images in Food-11. Results show that using GoogLeNet features used to train conventional machine learning algorithms can achieve higher results than a fine-tuned GoogLeNet model for detecting food images in Food-11. These results illustrate the convenience of using deep learning with machine learning classifiers through deep feature extraction as the user does not need to use a powerful GPU to train an efficient image classification model quickly. Many deep learning packages such as Tensorflow and MatConvNet give users the ability to fine-tune CNNs using CPU, however, it has been stated that using a GPU can be around 8 times faster than using a CPU in training a CNN [269].

Food-5K AUC results achieved in this work were close to 1 in validation and evaluation image sets using ANN and RF with both ResNet-152 features and GoogLeNet features. However, the validation and evaluation test sets are small in comparison to other popular food image datasets with only 500 in each class for each dataset, and therefore more research is needed in classifying a more extensive range of food images types and image quality. Food-5K training dataset, which was used to train food/non-food models, is also comparatively small with 1500 images in each class and contains limited food image types. Therefore further research would need to be completed in training machine learning classifiers with a diverse food image training dataset. Further evaluation was completed using the food/non-

food trained models that achieved highest accuracies with Food-5K to classify a new image dataset that combines food images in UNICT-FD889 and non-food images Caltech-101, called UNICT-Caltech, which is larger than the validation and evaluation sets provided in Food-5K [260, 261] containing 3583 food images and 9144 non-food images. Results from classifying this dataset are listed in Table 6.11 and show that with using Food-5K training dataset to train machine learning classifiers can achieve a high food accuracy using SVM-RBF achieving 97.50%.

Further experiments focused on using deep features to classify food texture image items under different illuminations, and authors of RawFooT-DB also researched the use of using other popular pretrained CNNs for feature extraction (AlexNet, VGG networks). The experiments presented in this work utilised deep residual network features and GoogLeNet features to classify food images in different lighting settings. Other research that used RawFooT-DB [20] divided the food image classes into illuminant categories. In this work, we evaluated the performance of ResNet-152 features in classifying food texture images across a range of different lighting conditions.

Results from using ResNet-152 to train an ANN achieved 99.28% accuracy, and a ROC value of 0.99 and the same features with SVM-RBF achieved 99.10%. More importantly, the use of deep features with supervised machine learning algorithms, using ResNet-152 and GoogLeNet, can generalise between food texture types with great efficiency under different illuminations. Results from RawFooT-DB support results in early experiments in that ResNet-152 features marginally outperform GoogLeNet features even in determining food classes across some illuminations and low in-between class variance. Figure 6.15 states the performance of classifying each texture class in RawFooT-DB using GoogLeNet features with ANN, and similar decreases in F-measures are present when compared to ResNet-152 ANN and SVM-RBF in Figure 6.13 and 6.14. GoogLeNet features also experienced

misclassifications with white peas and chickpeas, and with several meat textures (salami and hamburger).

Results show that most experiments with RawFooT-DB using both feature types achieved over 90% accuracy (apart from Gaussian Naive Bayes experiments). There were several misclassifications between similar food groups. The food textures that were misclassified are very alike in texture and shape (chickpeas and white peas), and the images used for testing and training are focused on the food texture without the overall food item shape and size as shown in Figure 6.10 and 6.11. The use of a texture based classification model trained using deep features can also be incorporated into a semi-automation approach to food logging in which the user can manually segment food portion for classification. Future work could enable the user to utilise a polygonal tool to draw around the food item and then a food texture based classifier can you used to predict the food item thus removing much of the complexity and noise of other food and non-food items in the food image.

Figure 6.16, 6.17, and 6.18 depict ascending F-measure scores of each food class in RawFooT-DB texture class. The F-measure for ResNet-152 and GoogLeNet features used to train SVM-RBF and ANN yield high accuracy results for all classes, with no classes falling below 0.90 F-measure. However, several classes that did achieve lowest F-measure appear in both ResNet-152 and GoogLeNet trained model results. Table 6.19 shows the food classes that achieved lowest F-measure for the models that achieved highest overall classification accuracy. Much of the food classes listed in Table 6.19 share common characteristics. Kernel-based foods such as chickpeas, white peas, and currants are listed that share similar characteristics (Figure 6.10). Milk chocolate and chicken breast texture images patches in RawFooT-DB contain low in-between class variance and appear in lowest F-measure scores, however, the classes mentioned in Table 6.19 still achieve over 0.90 F-measure using deep

features from both ResNet-152 and GoogLeNet and misclassification count total for these classes are small compared to results listed in Chapter 5.

Table 6.19 Ten Classes that achieved lowest F-measure in each RawFooT-DB models that achieved highest accuracy.

RawFooT-DB Model	**10 lowest classes (based on F-measure)**
ResNet-152 + SVM-RBF	chickpeas, white peas, chicken breast, mango, apple slice milk chocolate, swordfish, air-cured beef, sultana, hamburger
ResNet-152 + ANN	white peas, chicken breast, chickpeas, milk chocolate, tuna swordfish, sultana, hamburger, salmon, salami
GoogLeNet + SVM-RBF	hamburger, sultana, salami, pumpkin seeds, currant, chicken breast white peas, pomegranate, chili pepper, chickpeas

It was revealed that ResNet-152 features achieved higher classification accuracy results when compared to GoogLeNet. Therefore, ResNet-152 was used to classify Food-101 dataset. The images in Food-101 were not developed in a controlled environment but collated using a social media website (Foodspotting), which were uploaded by users and taken in real-world situations (restaurants, at home, cafes, etc.). The images are also captured under illuminations, and the dataset contains image quality of the images vary, and no bounding box information is provided to help determine where the food items are located in the image. Food-101 comprises of 101,000 images and 1,000 for each food class, and because of the size of this dataset, we partitioned dataset in training and validation using 75:25 ratio, 75% used for training and 25% used for testing and used a random state of '1' with the scikit-learn library. The highest accuracy achieved using ResNet-152 deep features extracted from Food-101 was 64.98% using an SVM with RBF kernel using ResNet-152 features. The full breakdown of results using ResNet-152 to classify Food-101 are located in Table 6.16. The features extracted from layers deep in CNN architecture provide efficient representations

that can be used to classify even the most challenging food image datasets such as Food-101. The quality of food images present in Food-101, in regards to food variation and noise, i.e. other non-food items, and unrelated food items, may be a factor in the decrease in accuracy. Comparing the results of Food-101 (101 classes) with RawFooT-DB texture dataset (67 food classes) suggest that the class size may not a major determining factor in the decrease in accuracy but the quality of the images used in regards to being genuinely representative of the class. Results achieved in this work in classifying RawFooT-DB is comparable with results obtained in [20] albeit the authors created small subsets for each lighting condition, while work presented in this Chapter extracted features from each food class that contains a variety of lighting conditions.

Table 6.20 Ten Classes that achieved lowest F-measure in Food-101 for ResNet-152 + SVM-RBF model

Food-101 Model	**10 lowest classes (based on F-measure)**
ResNet-152 + SVM-RBF	steak, foie gras, bread pudding, tuna tartare, scallops, omelette pork chop, huevos rancheros, apple pie, crab cakes

Table 6.20 lists the food classes that achieved the lowest F-measure with ResNet-152 and SVM-RBF model. Steak and foie gras achieved lowest F-measure with 0.36 and 0.37 respectively. Steak food class was frequently misclassified with other meat dishes such as pork chop, prime rib, foie gras, and filet mignon and similar misclassifications were reported for foie gras (Figure 6.24). Foie gras experienced similar misclassifications to steak (Figure 6.25). Bread pudding was also misclassified as other dessert dishes such as Pana cotta, apple pie, tiramisu, and strawberry shortcake (Figure 6.26). Tuna tartare was also classified as other seafood dishes such as sushi, and crab cakes, ceviche as well as other meat dishes such as beef tartare and foie gras. Tuna tartare was also classified as several desserts such as Pana

cotta and chocolate mousse as well as several salad dishes such as beet salad and Caprese salad (Figure 6.27). From the example images listed in Figure 6.24-27, there are significant variations in shapes and colours amongst each class, and because Food-101 is a food image dataset that contains images from a free-living environment, much of the classifications share characteristics in regards to non-food items related objects, e.g. plates. Therefore for accurate deep feature extraction using pretrained CNNs, the food item must be isolated, and images must be cleaned for more precise classification. This is evident in RawFooT-DB results in which both pretrained CNNs were able to extract relevant deep features from isolated texture patches and experiments were shown to achieve high performance even in classifying food classes with minimal in-between class variance.

Fig. 6.24 Example of classes classified as **steak** class in Food-101 using ResNet-152 with SVM-RBF.

Fig. 6.25 Example of classes classified as **foie gras** class in Food-101 using ResNet-152 with SVM-RBF

Fig. 6.26 Example of classes classified as **bread pudding** class in Food-101 using ResNet-152 with SVM-RBF.

Fig. 6.27 Example of classes classified as **tuna tartare** class in Food-101 using ResNet-152 with SVM-RBF.

For further comparison, Table 6.21 lists results achieved in this work with other research that used related deep feature extraction in classifying food image datasets. It is clear from Table 6.21 that ResNet-152 deep features echo results achieved with other datasets and other deep feature types in achieving similar accuracy results [45]. ResNet-152 deep features are able to achieve high classification accuracy in both fine grained datasets such as RawFooT-DB and binary decision datasets e.g. Food/NonFood, however there is a decrease in accuracy when food image datasets with high food variance and noise is present in images as seen in Food-101. A semi-automated approach could be applied to CNN deep feature classification that allows the user to draw around a food image before classification to remove noise, further analysis is needed to evaluate this approach and to measure improvement in accuracy.

Table 6.21 Summary of research using deep feature extraction to classify various food image datasets. **Bold** denotes results achieved in this work.

Extraction Model	Accuracy	Food Classes	Dataset
VGG-S [214]	92.47%	2 (Food/NonFood)	RagusaDB
NIN	90.82%	2 (Food/NonFood)	
AlexNet	84.95%	2 (Food/NonFood)	
GoogLeNet [220]	94.67%	2 (Food/NonFood)	Based on RagusaDS
	99.01%	2 (Food/NonFood)	FCD
NIN [212]	95.1%	2 (Food/NonFood)	IFD
Singla, et al. [209]	99.2%	2 (Food/Non-Food)	Food-5K
	83.6%	11	Food-11
AlexNet [215]	94.01%	7 (food groups)	PFID
	70.13%	61	PFID
AlexNet [216]	57.87%	100	UEC-FOOD100
AlexNet	70.41%	101	Food-101
AlexNet	78.77%	100	UEC-FOOD100
AlexNet	67.57%	256	UEC-FOOD256
VGG-19 [218]	40.21%	101	UMPC-Food-101
ResNet-152	**98.8%**	**2 (Food/NonFood)**	**Food-5K (Evaluation set)**
ResNet-152	**99.4%**	**2 (Food/NonFood)**	**Food-5K (Validation set)**
ResNet-152	**91.34%**	**11**	**Food-11 (Evaluation set)**
ResNet-152	**99.28%**	**68**	**RawFooT DB)**
ResNet-152	**64.98%**	**101**	**Food-101**

Using CNN deep features to classify food images datasets exceed the performance compared to other conventional feature selection methods and has been well documented [214, 215, 216]. Hand crafted feature selection methods such as SURF, or colour can encounter difficulties when classifying fine-grained classification of food categories as some public food image datasets contain small in-between class differences amongst large number of classes (e.g. Food-101). It has been stated in [51] that deep CNN features should be the first initial method for visual classification tasks due to their high performance in generalising to other datasets as CNNs are trained to be able to learn rich representations from a large number of images. CNNs able to determine complex filters to combine them with other patterns for greater detail. CNNs are able to produce internal image feature representation, which is advantageous when compared to hand crafted feature types such as SIFT, SURF or HOG. In this work, ResNet-152 features are able discriminate effectively between food and non-classes and in classifying high level food groups (Food-11), when compared to other works in [13]. It is clear that using ResNet-152 pretrained model is able to capture relevant image features to enhance the generalisation between fine-grained objects as demonstrated in classifying RawFooT-DB in Table 6.15 . ResNet-152 contains 152 layers that combine multiple convolutional and pooling layers to filter important image features and the use of residual connections to train the network produce accurate features can be highlighted for effective generalisation effectively to other datasets.

It is clear that using CNN features can enhance the accuracy of food image classification when compared to traditional feature extraction methods and this has been observed in other works, for example in [213] SURF and LAB colour features, and Random Forests were used to classify Food-101 dataset and achieved 50.76% accuracy. In [212] an AlexNet model was fine-tuned using food image categories, and deep feature extraction was performed after to classify Food-101, and authors achieved 70.41%, which is a significant increase when

compared to results obtained in [213]. As well as deep feature extraction, fine-tuning was also used to classify Food-101 and authors in [48] achieved a top-1 accuracy of 77.4% after 250,000 iterations in training a CNN architecture called 'DeepFood', which is a significant accuracy increase in comparison to [213]. In [218] fine-tuning was also used to classify Food-101 dataset was also used to fine-tune Inception V3 architecture and achieved a top-1 accuracy of 88.28%. Research in [212] also achieved a top-1 accuracy of 65.32% using HOG features, colour values with fisher vectors in classifying UECFOOD-100, however, CNN based features extracted from a modified AlexNet model with a linear SVM achieved an increased accuracy of 78.77%. For UECFOOD-256 dataset, work presented in [270] achieved a top 1 accuracy of 50.1% using HOG features and colour features with Fisher Vector representations and the same authors in later research [212] utilise deep CNN features extracted from a modified AlexNet and achieved a top 1 accuracy of 67.57% in also classifying UECFOOD-256 dataset. For RawFooT-DB food texture, dataset experiments were completed in classifying food textures under various lighting conditions, authors compared traditional feature extraction techniques with CNN based features, and results show that OCLBP and Gabor features achieved 95.9% and 96.2% accuracy respectively with deep CNN features achieving 98.2% accuracy [259]. From the literature, it is clear that using CNN deep feature extraction and fine-tuning can achieve superior results in food image classification.

6.6 Key Findings

The work presented in this study utilises state-of-the-art pretrained CNNs for deep feature extraction for food image classification. ResNet-152 and GoogLeNet pretrained CNN were selected and used for deep feature extraction with a variety of public food image datasets. A collection of machine learning algorithms were selected to use with extracted deep features. All machine learning classifiers trained using ResNet-152 and GoogLeNet

deep features are suitable for detecting food in photographs, however ResNet-152 deep features achieve higher accuracy than GoogLeNet with Food-5K, Food-11, and RawFooT-DB food image datasets. ANN and SVM models achieved highest percentage accuracy using ResNet-152 and GoogLeNet deep features in all food classification experiments. ResNet-152 consistently achieves higher percentage accuracies compared to GoogLeNet deep features in classifying food images. Combining ResNet-152 deep features with conventional machine learning classifiers is able to achieve similar results in comparison to other works that utilise fine-tuning to train pretrained CNN [209].

6.7 Implications for Dietary Management

The methods presented in this study highlight the performance of using state-of-the-art pretrained CNN for deep feature extraction to train conventional machine learning classifiers. It is clear from the results presented in this study that using ResNet-152 features to train food image classifiers achieve high accuracy results, particularly in detecting food in images and classifying high level food groups. In regards to dietary management implications, the methods presented in this work highlight the efficiency of using pretrained CNN for deep feature extraction for food image classification, results for Food-5K, Food-11, and RawFooT-DB suggest that deep feature extraction approaches can be used to train food image classifiers to detect food, food groups, and food texture patches. Percentage accuracy results for Food-101 were lower than that of previous results for other datasets due to size of Food-101 and the complicated nature of the dataset, however a reasonable accuracy was attained of 64.58%. ResNet-152 deep features are able to highlight important features that can be used to generalise across diverse food groups, especially food groups with low-in-between variance, this is evident from the classification accuracy results of RawFooT-DB in which ResNet-152 features were able to accurately discriminate between very similar food texture classes. The

results using RawFooT-DB also reinforce the need for a semi-automated approach in which the user is able to segment the food item to isolate the food texture. The use of a semi-automated approach may increase user interaction and also remove computational complexity (automatic food image segmentation). Also, for automated food logging, experiments presented in this Chapter show that deep features extracted using pretrained CNNs are able to determine food class in images captured under different illuminations and lighting conditions. ResNet-152 also achieve moderate accuracy results in classifying difficult free-living environment food images (Food-101 food image dataset). Results for Food-101 dataset with ResNet-152 deep feature extraction are promising and the methods outlined in this Chapter could be combined with the approaches outlined in Chapter 3. For food image classification applications, the use of deep ResNet-152 features combined with conventional machine learning algorithms can be effectively used in food dietary management web and smartphone applications without the need of specifying a unique CNN architecture and training from scratch. It is clear from the experiments presented that the use of deep feature extraction can be used to automate food logging for dietary management.

6.8 Summary

Image classification approaches have been applied to food image logging and traditionally conventional feature extraction methods have been used, however deep learning based approaches such as CNN have achieved greater accuracies (Chapter 2 Literature Review). This study presents experiments that utilise deep learning feature extraction approaches for food image classification. Deep features were extracted from a pretrained CNN ResNet-152 model initially trained using ILSVRC dataset to train supervised machine learning algorithms classify a variety of food image datasets; Food-5K, Food-11, RawFooT-DB, and Food-101. ResNet-152 was benchmarked against pretrained CNN GoogLeNet for performance compari-

son discussed in [209]. Deep features were extracted from each CNN model and used to train machine learning classifiers. Results show that ResNet-152 performed slightly better than GoogLeNet CNN model. Further comparison with other works suggest that deep feature extraction is able to achieve similar results without the need of fine-tuning a CNN model in regards to Food-5K and Food-11. However, further work is needed to fully evaluate both CNN approaches using other food image datasets. ResNet-152 deep features also achieve high accuracy results when generalising between high level food groups and also when classifying food texture image patches (RawFooT-DB) and ResNet-152 consistently achieves higher accuracy results when compared to GoogLeNet deep features. For Food-101, ResNet-152 achieved 64.98% in classifying 101 food classes. The decrease in accuracy in comparison of the other food datasets is based on the composition and acquisition of food images in Food-101. Food-101 is made up of images photograph in real-world environments and therefore may contain noise and high colour and texture variation. The high colour, texture, shape variance may contribute small-in between difference amongst classes. Experiments presented in this Chapter for food detection show that a range of machine learning classifiers are suitable, however SVM and ANN consistently achieved the highest accuracies, similarly with high level food classification (Food-11). ResNet-152 deep feature extraction is able achieve high accuracies using SVM and ANN classifiers in classifying Food-5K, Food-11, and RawFooT-DB (98.8%, 91.34%, and 99.28% respectively). Performance decreases considerably when classifying Food-101 with deep residual features (64.98%), deep feature extraction encounters difficulty when applied to difficult food image datasets there a fine-tuned CNN approach or CNN 'trained from scratch' may be more suited when classifying difficult food image datasets, further experimentation with other CNN architectures is needed.

Chapter 7

Discussion & Future Work

7.1 Introduction

This chapter discusses and evaluates the main findings, methods and achievements presented in this thesis. The work presented in this thesis focuses on exploring methods to use for food image classification for food logging and what information can be derived from images to remove much of the complexities of traditional food logging techniques. Another contribution of this thesis is exploring statistical approaches for calorie estimation using photographs of meal items. In Chapter 2 Literature Review, various technologies were discussed that have been applied in researching automated food image logging. Chapter 2 explored how food logging is a valuable tool to help reduce the risk of chronic conditions, The number of individuals that are obese is increasing, and food logging is a method that has been shown to help individuals lose weight and maintain healthy a diet through consistent food monitoring. Current limitations of traditional food logging techniques such as using a mobile-based device allow users to input the food item to determine calorie content based on using a nutritional database or API. Research on using mobile-based food logging techniques for dietary management suggests increased adherence due to convenience and mobility.

However, drop-off rates and low retention rate is still an issue. To increase adherence and usability of dietary management applications, focus has turned to using food images as a way to mitigate limitations of traditional methods and to increase adherence. In this thesis computer vision and image processing methods along with statistical approaches have been researched and applied to food image logging in the aim to increase convenience through an interactive and automated approach.

Using food images for food logging allows the user to ascertain contextual information regarding food portion size, and the number of different food types present in their meals when compared to traditional methods of food logging. Food images can serve as a visual aid to allow the user to remind themselves of what they consumed instead of recalling nutritional intake from memory. Research has been completed that has shown that food memory recall can lead to inaccuracies, and calorie underestimation and other research discuss the advantages of using food images to not only serve as a visual reminder but to give clarity and to enhance understanding. Computer vision methods have been applied to automate food logging through food image classification and have the potential of removing much of the complexity of traditional food logging methods. Figure 7.1 describes an automated food image classification pipeline; (1) the user photographs a food image and the image is processed for feature extraction in (2), which also may lead to image segmentation, in (3) image features extracted from the image are then classified using trained machine learning classifiers. The label is then outputted in (4), and nutritional information is then calculated in (5) and finally outputted to the user (6). The work presented in this thesis is related to specific sections within this automated food image logging pipeline.

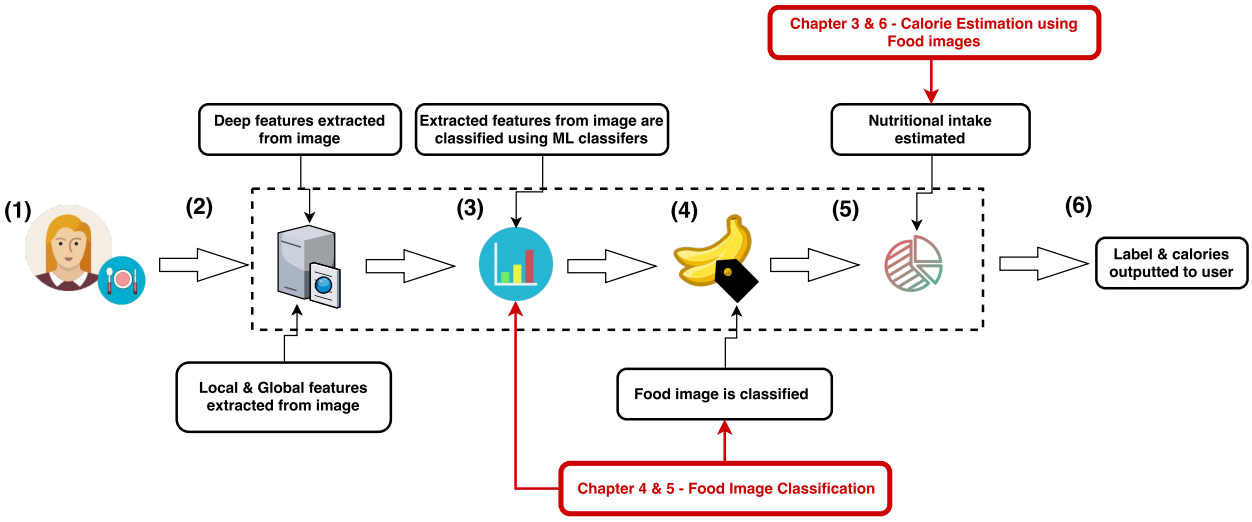


Fig. 7.1 High level system design to classify images and to estimate calories for food logging.

7.1.1 Research Aim

The central aim of this work was two fold: 1) to investigate, develop, and evaluate computer vision and deep learning approaches for food image classification and 2) To explore and evaluate methods that could be used for predicting nutritional content in food images. The following objectives were identified to help achieve this aim:

7.1.2 Research Objectives

- To identify methods to quantify calorie content of food items in pictures using image processing methods.
- To examine the use of crowdsourcing approaches to predict nutritional content of food images for dietary management.
- To examine the use of feature extraction approaches with supervised machine learning algorithms that could be used for food image classification to promote food logging.
- To examine the effects of using feature fusion for food **detection** in images.

- To examine the effects of using feature fusion to **predict** specific food items in photographs captured in free-living environments.
- To examine the effects of using feature fusion to predict image texture patches of food items.
- To examine the use of state-of-the-art pretrained deep learning models for deep feature extraction to classify food portions in images.

7.1.3 Research Questions

The following research questions were also identified using the Literature Review presented in Chapter 2 and the remainder of this section will discuss each research question and how they relate to specific work Chapters.

1. What state-of-the-art food photography logging techniques are currently used for dietary management? (Chapter 2)
2. How can calories be calculated from a food portion in a photograph through correlating calories with pixels? (Chapter 3)
3. How can crowdsourcing be utilised to support accurate calorie prediction to support dietary management? (Chapter 4)
4. What computer vision and deep learning approaches are available for food image classification for automated food logging? (Chapter 5)
5. How accurate are deep feature extraction approaches using pretrained CNNs compare to conventional feature extraction approaches in predicting food items in images? (Chapter 6)

The work Chapters presented in this thesis focused on answering research questions (2), (3), (4), and (5) in evaluating image processing methods and crowdsourcing approaches

for calorie estimation and using feature extraction, machine learning techniques for food image classification. The literature review presented in Chapter 2 provided an opportunity to research what current feature extraction methods that have been used for automatic food logging and the limitations of various feature extraction methods in classifying food types. It is clear from the literature review presented in Chapter 2 that there has been some success in applying traditional computer vision methods for food logging to provide for an automated workflow for food image recognition and nutritional calculation. The remainder of this Chapter discusses the research presented in this work and how it achieves the aim and objectives highlighted in Chapter 1. Figure 7.2 is a summary of what feature extraction methods, machine learning algorithms, and calorie estimation methods have been applied in an automated food image logging pipeline.

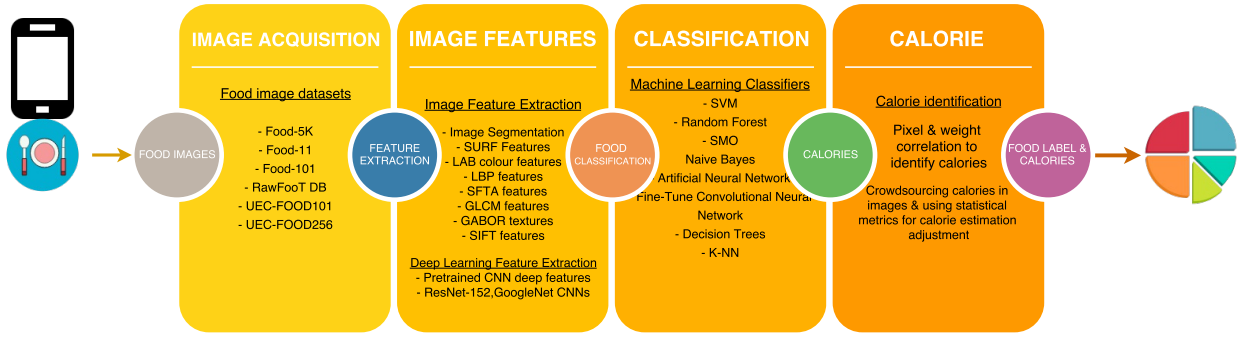


Fig. 7.2 Approaches that have been used in previous research for automated food image logging.

Figure 7.2 highlights a number of different approaches that could be employed for feature extraction methods and types that could be used for food image classification as well as various machine learning classification. Research questions identified are concerned with 2 areas of automated food logging; (1) food image classification and (2) statistical and image processing methods to determine nutritional content in images. The work presented in this thesis have answered the research questions through presenting approaches to automated food logging by combining a range of existing approaches. Food image classification is the

first step in any automated food logging system and in this thesis computer vision approaches were investigated, traditional feature extraction approaches and pretrained CNN for automatic deep feature extraction. Results discussed in this thesis highlight that deep feature extraction using pretrained CNN combined with conventional machine learning classifiers outperform traditional feature extraction approaches (Chapter 5 and Chapter 6), and these results also support results of other published research (Chapter 2, 4). Once the food portion is classified, the second step is nutritional estimation and in this thesis image processing, statistical, and crowdsourcing approaches have been investigated and evaluated in predicting calories in photographs of food (Chapter 3 and Chapter 4). The remainder of this Chapter will discuss the achievements and contributions to knowledge each study presented in this thesis and to propose a dietary management framework that incorporates elements of each study for food image logging to promote dietary management. The remainder of this Chapter will discuss contributions this thesis has achieved across each work Chapter.

7.1.4 Contribution to Knowledge

The following contributions to knowledge were achieved in 2 areas: food image classification and food image calorie estimation for food logging.

7.1.5 Research Contribution

The following research contributions were achieved in this PhD project relating to food image classification and calorie estimation to support dietary management.

- Proposed dietary management system based on RFPM that combines deep learning approaches for food image classification and crowdsourcing for calorie estimation.
- Proposed a calorie calculation pipeline that combines user interaction for image segmentation and image processing methods with linear regression.

- Crowdsourcing calorie adjustment approach was proposed to promote accuracy in food logging for dietary management. Experiments completed that suggest that calorie crowdsourcing adjustment approach improves accuracy of food logging.
- Utilised feature extraction methods and feature fusion with Speeded-Up-Robust-Features (SURF), Segmented Fractal Texture Analysis (SFTA), LAB colour features, and local binary patterns (LBP) with supervised machine learning techniques to classify food image categories.
- Texture feature fusion was used (SFTA and LBP features) for image classification of isolated food image textures and achieves similar results to other state-of-the-art CNN deep feature extraction approaches.
- Combined CNN ResNet-152 deep feature extraction with machine learning classifiers to classify variety of food image datasets. Results achieves higher accuracy in determining food group types in comparison to other published works.

The remainder of this section will discuss each contribution and how it was achieved based on methods and results from each work Chapter.

7.2 Discussion

- **Proposed a calorie calculation pipeline that combines user interaction and image processing methods with linear regression. - (Chapter 3)**

  The work described in this Chapter 3 discusses the use of semi-automation to accurately predict calorie content within a food item portion. The work explored in Chapter 3 entitled 'Semi-Automated System for Predicting Calories in Photographs of Meals', that combines image processing methods with linear regression to determine

calories in segmented food portions. The developed application was tested using various food portions of the same food item and was compared with ground truth calories to measure performance. Results show that the application can ascertain the calorie content with an average percentage error of 11.82%. The error percentage for each portion accounted for only a small number of calories due to the small number of calories already existing in the food item. The purpose of this work was to address accuracy issues of users self-reporting the nutritional content of their meals. In many food log applications, the user must recall the food item as well as the portion size (cup, tablespoon, gram). Many users may not know the exact portion size is on their plate, which may lead to inaccuracies and wrong data being recorded. The developed system allows users to use a tool to manually draw around a food portion to calculate the area (ensuring that the reference point is placed next to the plate to give context awareness). The system then outputs an approximation of the portion's calorie content. This process also addresses the inaccuracy issue in regards to portion size. This work highlights the importance of logging accurate food intake in the battle against obesity. Accurate calorie identification is vital in allowing individuals to manage dietary intake to promote healthy living. This process discussed in this Chapter 3 demonstrates the method of correlating area of food portions with calories to develop a regression model to predict calories in image images for food logging. Key messages from Chapter 3 proposes a method in which calories can be determined through correlating food area with calories combined with a context reference point to calculate the area of food portion. The study presented in Chapter 3 also highlights the use of manual image segmentation to calculate calories, the reason for using a manual approach because it will allow users to accurately pinpoint the size of the food portion and also to remove computational complexity. Chapter 3 also presents a method in which calorie estimation is personalised as opposed to using an online API where the user often has

to estimate the weight of the food item. Results of experiments suggest that using area of food for specific food items can calculate calories with a mean percentage error of 11.82% using the regression analysis for the food type used in Chapter 3. Further experiments could be completed to improve results by collecting a larger ground truth dataset through using 5 gram weighted portions as opposed to 10 gram weighted food portions. This work could also be extended to determine if the approach of correlating pixel values with weighted portions can be applied to other food types.

- **Crowdsourcing calorie adjustment approach was proposed to promote accuracy in food logging for dietary management. Experiments completed that suggest that calorie crowdsourcing adjustment approach improves accuracy of food logging. - (Chapter 4)**

  Results from this Chapter 4 suggest that crowdsourcing provides a novel approach for calorie estimation and adjustment. The aim of this work was to investigate the feasibility of utilising crowdsourcing as a method to provide dietary management for food logging. Analysis of the collected data from the online survey showed that expert group achieved high accuracy in determining calorie content of different meal images using different statistical metrics. The expert group also achieved greater accuracy in comparison to the non-expert group. The descriptive statistics generated from each group dataset each revealed differences between the group calorie prediction performance, the variance of estimations (standard deviation) was much smaller for each meal image for the expert group. For the non-expert group, there is a greater standard deviation, particularly with meals with a higher calorie ground truth. Correlation coefficient tests were completed on non-expert dataset using the ground truth calorie and the standard deviations for each meal image. Result show documented in Chapter 4 shows that there is a correlation between the ground truth calorie and the standard deviation of the meal images. Other experiments compared crowdsourced

calorie differences compared to individual calorie differences and results show that using the mode (calculated using individual calorie estimations for each meal) was more accurate than 50.28% of individuals based on mean calorie differences (Chapter 4, Table 4.4). These results further indicate the potential of using crowdsourcing for dietary management as experiments show using a group of participant reduces overall calorie estimation error in comparison to individual participants.

Secondary experiments consisted of using crowdsourced statistical metrics to adjust user calorie estimations with the aim of increasing the accuracy of food logging. This approach described in Chapter 4 details how a user's calorie prediction of a food image can be adjusted using overall calorie differences computed from a training dataset. To evaluate this process, the non-expert dataset was used instead of the expert dataset. This was due to the non-expert dataset being much larger dataset and allowed us to evaluate the calorie adjustment process using 5-fold cross validation fully. The mean calorie difference was computed using the training dataset meal calorie estimations and the ground truth calories. An overall mean calorie difference was used to deduct calories from estimations in the test dataset. A calorie baseline was also computed using each training fold, and this baseline was a determinant on whether to deduct the overall mean calorie difference computed from training folds. Results using this method revealed a reduction in overall calorie difference across user estimations. Results noted in Chapter 4 show that this novel approach has the potential to improve calorie adjustment across user food calorie predictions. Further evaluation of this approach shows a reduction in percentage error in the majority of images. The percentage error was computed using the ground truth and original mean calorie estimations. This process was repeated using the ground truth calorie and the new adjusted mean calorie estimations. The full breakdown of these results is noted in Chapter 4 Results section. The study presented in Chapter 4 suggests that

crowdsourcing calorie estimation metrics of non-experts can be used to adjustment future calorie estimations to enhance accuracy.

- **Utilised feature extraction methods and feature fusion with Speeded-Up-Robust-Features (SURF), Segmented Fractal Texture Analysis (SFTA), LAB colour features, and local binary patterns (LBP) with supervised machine learning techniques to classify food image datasets. - (Chapter 5)**

  Chapter 5 uses a feature combination approach to train several machine classification models. The primary contribution of this work was to investigate the use of a feature fusion approach using a combination of feature extraction types for food image classification. A variety of machine learning classification methods were also used to determine the most efficient combination of feature types. Three food image datasets were used in Chapter 5, Food-30, RawFooT-DB, and Food-5K. Food-30 and Food-5K contained images that were captured in free-living environments and RawFooT-DB contained images captured in a controlled environment. The performance of the feature fusion approach was assessed using a 10-fold cross-validation approach and dedicated evaluation datasets were used for RawFooT-DB and Food-5K .

  Various feature types were extracted from Food-30, RawFooT-DB, and Food-5K food image datasets, and several machine learning classifiers were selected to train food recognition models. Feature combination approaches were used to train each machine learning classifier to determine the optimal feature fusion approach.Food-30 was the most difficult dataset among the 3 datasets due to image acquisition method and the amount of food types. BoF was used to build a visual vocabulary to determine a feature vector for SURF and LAB colour features. For Food-30, various sizes of visual vocabularies were used to determine the optimal feature dimension for each feature type. Results show that a feature combination approach enhances the performance of food image classification to classify food images taken in real-world environments and

not prepared in a laboratory setting. When feature fusion approach was adopted the accuracy improved considerably particularly when SURF and LAB colour features were combined. Results show that an ANN trained with BoF SURF, BoF colour, SFTA, and LBP achieves accuracy result of 69.43% accuracy for Food-30. ANN consistently achieved higher accuracy across all feature combinations, and Naive Bayes achieved the lowest accuracy in each feature combination test. Across all experiments, SMO and ANN machine learning classifiers achieved highest accuracies in classifying food images in free-living environments.

The results of this work for Food-30 are comparable with other related research in classifying food images captured in free-living environments. It is important to note that the images used in Chapter 5 were not segmented, but the entire image was used for feature extraction. From the experiments, it is revealed that a reasonable degree of accuracy can be achieved through classifying non-segmented meal images using conventional feature extraction approaches. This work shows that there is potential to utilise conventional based feature extraction and machine learning classifiers to classify entire food meal images with reasonable accuracy. This work further highlights the need for a feature fusion approach to classifying challenging classes with small in-between class variance.

- **Texture feature fusion was used (SFTA and LBP features) for image classification of isolated food image textures (RawFooT-DB) and achieves similar results to other state-of-the-art CNN deep feature extraction approaches - (Chapter 5)**

  In regards to RawFooT-DB, experiments revealed were that SFTA and LBP features achieve similar accuracies to other state-of-the-art computer vision approaches (CNN based approaches). ANN and SMO achieve highest percentage accuracy in classifying RawFooT-DB texture images using a combination of LBP, SFTA, and SURF features. Experiments indicate that SFTA and LBP features can achieve high

percentage accuracies in classifying food texture images captured in a variety of different lighting conditions. When SURF features were combined with SFTA and LBP, percentage accuracy marginally increased to 93.91% (23,500 correctly classified food images) from 92.38% (23,118 correctly classified food images) using ANN in classifying 68 food texture classes. These results suggest that a feature fusion can marginally increase accuracy when classifying isolated food textures. This is a small percentage accuracy increase and more research is needed in refining the hyperparamters of SFTA feature extractor and LBP feature extraction process to determine optimal feature combination approach. For Food-5K, ANN and SMO also achieved highest percentage accuracies using a feature combination approach. The combination of conventional feature extraction approaches (SURF, colour features, SFTA, LBP) achieves high percentage accuracies with machine learning algorithms for food detection (ANN and SMO). Ten-fold cross validation show that using all features combined (training set) achieved highest percentage accuracy with 94.13% using an ANN. In regards to single feature types, SURF features with ANN and SFTA features with SMO achieved highest percentage accuracy with 88.63% and 88.8%. Validation and evaluation datasets achieved 95.8% and 91% accuracy respectively combining SURF, colour, SFTA, LBP features with ANN. All single feature types with each machine learning classifier achieved over 70% accuracy and experiments are promising in using conventional feature training for food detection. Experiments indicate that conventional features can be used to train food detection models provide were able to adequate generalise between food and non-food image classes in Food-5K. Food images contained in validation and evaluation Food-5K datasets are collected from real-world environments therefore the images may contain noise and high colour variation. Validation and evaluation datasets contain only 1,000 images and each class is balanced (500 food and non-food images), these datasets are small in comparison

to other datasets. Further testing is needed using the classification models trained in Chapter 5 using a larger evaluation dataset to allow for greater analysis.

- **Combined CNN ResNet-152 deep feature extraction with machine learning classifiers to classify variety of food image datasets. Results achieves higher accuracy in determining food group types in comparison to other published works. - (Chapter 6)**

  Chapter 6 discusses the use of convolutional neural networks (CNN) for food image classification. The use of CNN for image classification can be summarised into 3 area; training CNN from scratch, fine-tuning pretrained CNN, and CNN for deep feature extraction. Research was focused on transfer learning, i.e. using a pretrained CNN for deep feature extraction and using these features to train machine learning classifiers for food recognition. A literature review was completed that researched the power of pretrained CNN for feature extraction combined with supervised machine learning classifiers to classify food images for automated food logging. The literature review was able to identify what food image classification problems deep feature extraction has been applied to (food detection and food recognition) it is important to highlight that the literature review focused on research that used deep feature extraction to classify food images opposed to fine-tuning or training a new CNN architecture from scratch as deep feature extraction has proven to be efficient in fine-grained image classification. Instead of training a CNN using a GPU, pretrained models can be utilised for convenience without defining a CNN architecture or appending new layers to a pretrained for fine-tuning. Table 7.1 is a summary of the research that utilised deep feature extraction for food image classification. After literature review was completed, it was decided that ResNet-152 pretrained model would be used to extract deep features to train machine learning algorithms to classify a variety of diverse food image datasets.

Table 7.1 Summary of research using deep feature extraction to classify various food image datasets. **Bold** denotes results achieved in this work.

Extraction Model	**Accuracy**	**Food Classes**	**Dataset**
VGG-S [214]	92.47%	2 (Food/NonFood)	RagusaDB
NIN	90.82%	2 (Food/NonFood)	
AlexNet	84.95%	2 (Food/NonFood)	
GoogleNet [220]	94.67%	2 (Food/NonFood)	Based on RagusaDS
	99.01%	2 (Food/NonFood)	FCD
NIN [212]	95.1%	2 (Food/NonFood)	IFD
Singla, et al. [209]	99.2%	2 (Food/Non-Food)	Food-5K
	83.6%	11	Food-11
AlexNet [215]	94.01%	7 (food groups)	PFID
	70.13%	61	PFID
AlexNet [216]	57.87%	100	UEC-FOOD100
AlexNet	70.41%	101	Food-101
AlexNet	78.77%	100	UEC-FOOD100
AlexNet	67.57%	256	UEC-FOOD256
VGG-19 [218]	40.21%	101	UMPC-Food-101
ResNet-152	**98.8%**	**2 (Food/NonFood)**	**Food-5K (Evaluation set)**
ResNet-152	**99.4%**	**2 (Food/NonFood)**	**Food-5K (Validation set)**
ResNet-152	**91.34%**	**11**	**Food-11 (Evaluation set)**
ResNet-152	**99.28%**	**68**	**RawFooT DB)**
ResNet-152	**64.98%**	**101**	**Food-101**

Further work presented in Chapter 6 investigates the use of deep features extracted from pretrained CNNs for food image classification. The main contribution of these work is utilising ResNet-152 and GoogleNet pretrained CNN models for deep feature extraction to classify a wide variety of food image datasets. The main contribution of these work is two-fold; (1) utilising ResNet-152 and GoogleNet pretrained CNN models for deep feature extraction to classify a wide variety of food image datasets, and (2) comparing the use of pretrained GoogleNet with a fine-tuned GoogleNet model presented in [1] in detecting food/non-food images and food groups. In Chapter 6, 2 popular pretrained CNNs were compared, ResNet-152 and GoogleNet and extracted deep features from layers deep in each CNN architecture to classify several food image datasets; Food-5K, Food-11, and RawFooT DB. Each dataset used in these experiments was developed for different objectives; Food-5K was used to train food image detectors; Food-11 to determine high-level food groups; RawFooT-DB was used to classify isolated texture patches, and Food-101 consisted of specific food item images photographed in real-world environments. The aim of the work presented in Chapter 6 was to evaluate the performance of deep residual networks for feature extraction using the ResNet-152 pretrained model when compared to GoogleNet pretrained model. The first set of experiments focused on classifying Food-5K dataset which was developed by authors in [209] to explore computer vision methods for food detection. Food-5K dataset consisted of 2 categories; food and non-food, training is balanced and contains 1500 images of each category [209]. Food-5K dataset contains a validation and evaluation set, and each category contains 500 images each per dataset. The authors developed this dataset to measure the performance of fine-tuning GoogleNet pretained CNN for classification. Food-5K was developed by selected images from already publicly available datasets e.g. Food-101 [213], UECFOOD-100 [257] and UECFOOD-256 [258]. The authors describe this dataset as being varied

as they wanted to select foods that cover a wide variety of different food dishes. The images also contain some noise, and multiple food items may be contained in an image. The non-food images consisted of images that do not contain food items (objects or humans). Food-5K was used to find out how ResNet-152 deep features perform in detecting food items in images, which can be argued is an important first step in food image classification for food logging. The authors developed the non-food image dataset from using other publicly available datasets, e.g. Caltech101, Caltech256, Emotion6, and Images of Groups of People.

Results from using Food-5K show that deep residual features were able to detect food in images with high percentage accuracy across all machine learning classifiers, achieving over 90% accuracy in each experiment. We benchmarked our experiments using the results achieved by the authors of Food-5K and Food-11 datasets who used a fine-tuned GoogleNet [209] and these results in our work suggest that there is potential to achieve high accuracies and performance without the need of fine-tuning pretrained CNNs for particular datasets, and that deep feature extraction can also be used. Furthermore, due to the nature of Food-5K being a binary decision between food and non-food classes, generic deep features may be sufficient enough to provide adequate generalisation to classify between two classes (i.e. food and non-food). ANN and support vector machine (SVM)-RBF trained with ResNet-152 features achieved the highest accuracies in the majority of Food-5K experiments and the Food-5K ANN, and SVM-RBF model was further evaluated by classifying the entire Food-11 dataset for food detection. Results show that our ANN model trained using ResNet-152 features achieved higher food detection accuracy compared to the fine-tuned GoogleNet model in [209] when tested against Food-11 image dataset. We also evaluated both our Food/Non-Food SVM-RBF model trained with ResNet-152 and GoogleNet deep features using Food-11 for food detection and results showed that

these models achieve marginally higher results compared with other results achieved in [209].

Further experiments were completed in Chapter 6 that explored the use of residual deep features to classify fine-grained food images (RawFooT-DB) and food images (Food-101) photographed in real world environments. In these experiments RawFooT DB and Food-101 food image datasets were used. RawFooT DB contains isolated texture patches of various food items under different illuminations and lighting conditions and is partitioned into a training and testing split. The experiments in Chapter 6 extract deep residual network features and GoogleNet features from each image patch and use these to train machine learning classifiers. Results from using ResNet-152 to classify show that an ANN achieved 99.28% accuracy and and a ROC value of 0.99 and the same features with SVM-RBF achieved 99.10%. More importantly, the use of deep features with supervised machine learning algorithms, from both ResNet-152 and GoogleNet, are able to generalise between food texture types achieving high percentage accuracy across different illuminations. Results from RawFooT-DB echos results in early experiments in that ResNet-152 features marginally outperform GoogleNet features even in determining food classes across a number of illuminations. The results show that deep features from ResNet-152 and GoogleNet efficiently classify isolated texture image patches across various lighting conditions with high classification accuracy reported. For each experiment, ResNet-152 achieves higher performance compared to GoogleNet features, however it is worth noting that both deep features experience misclassification for similar food types as discussed in Chapter 6.

The images in Food-101 dataset were not developed in a controlled environment but collated using a social media website (Foodspotting), which were uploaded by users and taken in real world environments (restaurants, at home, cafes, etc.).The images are also taken under illuminations and the dataset contains image quality of

the images vary considerably and no bounding box information is provided to help determine where the food items are located in the image. Food-101 contains 101,000 images and 1,000 for each food class, and because of the size of this dataset, we partitioned dataset in training and validation using 75:25 ratio, 75% used for training and 25%.The highest accuracy achieved using ResNet-152 deep features extracted from Food-101 was 64.98% using an SVM with RBF kernel. The features extracted from layers deep in CNN architecture provide efficient representations that can be used to classify even the most challenging food image datasets such as Food-101. The quality of food images present in Food-101, in regards to food variation and noise i.e. other non-food items, and unrelated food items, may be a factor in the decrease in accuracy. Comparing the results of Food-101 (101 classes) with RawFoot texture dataset (67 classes) suggest that the class size may not a major determining factor in the decrease in accuracy but the quality of the images used in regards to being truly representative of the class. Results achieved in this work in classifying RawFooT DB is comparable with results achieved in [259] albeit the authors created small subsets for each lighting condition. Authors in [259] did create a subset which encompassed all lighting conditions, and work presented in Chapter 6 following these procedures to compare results. Results achieved in this work using deep CNN ResNet features achieved 99.28% using an ANN which was an improvement on the results achieved in [259] using VGG-16 deep features with 98.21% using a nearest neighbour approach. Key messages from this study show that ResNet-152 deep features combined with a variety of machine learning classifiers are able to detect food items in images with high accuracy (90% > accuracy across each experiment). Results from Chapter 6 also highlight the effectiveness of using pretrained ResNet-152 deep features to detecting food in images and classifying food images into high level food groups. ResNet-152 deep features achieve higher percentage accuracy than fine-tuned GoogleNet with

Food-11 in [209] and achieves comparable percentage accuracy with Food-5K also compared to GoogleNet [209]. Experiments completed in Chapter 6 suggest that ResNet-152 deep features provide generalisation power to determine between food and non-food classes and high-level food groups and results suggest deep feature extraction using pretrained ResNet-152 CNN and conventional machine learning algorithms have the ability to outperform fine-tuned models. In regards to classifying specific food types, ResNet-152 achieves comparable accuracy to other similar works in classifying food texture image RawFooT-DB and also images captured in free-living environments. Chapter 6 demonstrates that transfer learning methods such as deep feature extraction from CNN has the potential to accurately classify food items in images for automated food logging.

- **To propose a dietary management framework that combines elements from each Chapter for food image recognition and calorie estimation.**

  Research presented in this work can be combined to develop a framework for food image logging; the combination of crowdsourcing nutritional information and using deep CNN feature extraction for image classification. These technologies can be used to inform the development of a framework that can be used to promote accurate food logging by removing much of the complexity of traditional food logging. This use of semi-automation food item segmentation can be combined before deep feature extraction and classification to remove noise and other food items.

  Figure 7.3 illustrates a pipeline that describes the combination of different technologies presented in each study of this thesis. The purpose of combining these technologies is to present an automated food logging platform that utilises semi-automation for food logging. The user of such a system would upload an image captured on a smartphone device. The user is then able to use a polygonal tool to segment the food portion or use the entire image for food image classification. Machine learning

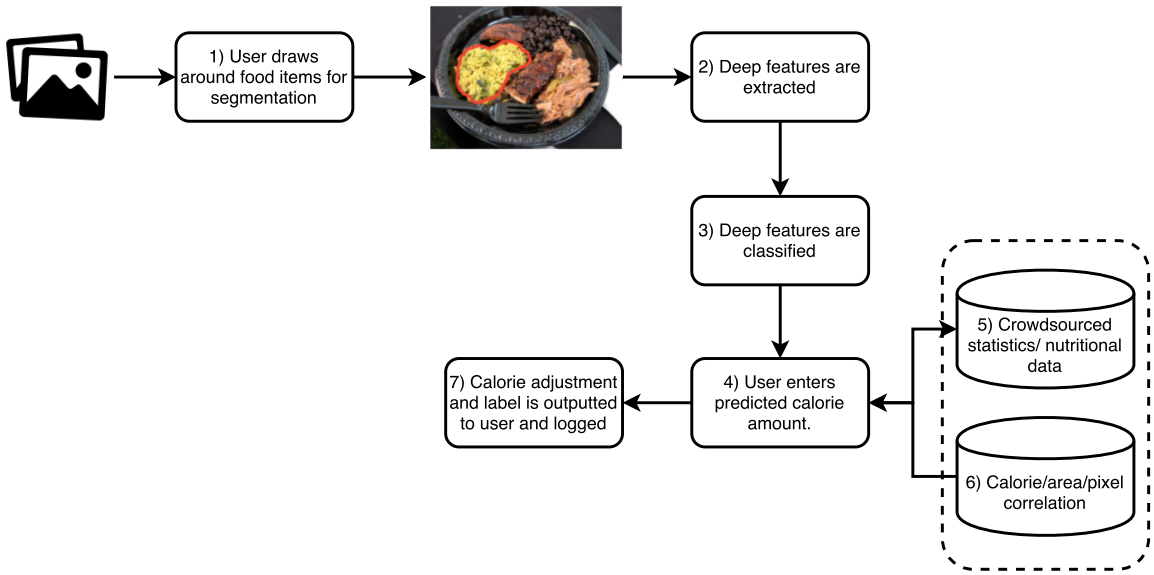


Fig. 7.3 Proposed system pipeline that incorporates different technologies and statistical approaches that allow for image classification and energy intake calculation.

algorithms are then applied to extract deep features using pretrained CNN models. This stage relates to Chapter 6 in which deep features were used to classify a range of diverse food image datasets and results from these experiments reveal that utilising deep features extracted from pretrained CNNs to train conventional machine learning algorithms can generalise between high level (bread, vegetables) food groups and specific food items. After classification, the food label is then outputted (4) to allow for energy intake calculation. In (5) and (6) the energy intake calculation is completed using crowdsourcing methods used in Chapter 4. Crowdsourcing nutritional intake, such presenting different crowdsourced calorie metrics calculated using user estimations. Secondly, a calorie adjustment process can be used as discussed in Chapter 4. For calorie adjustment, the user would input an estimated calorie amount for the food portion in question then the adjusted algorithm would adjust the amount accordingly based on crowdsourced metrics.

7.3 Limitations & Prospective Research Studies

This section will discuss limitations of the approaches used in this thesis and provide potential areas for future work. The potential works presented in this section seeks to extend work that has already been completed and will discuss other methods and technologies that could be applied in order to enhance the performance and results. The section will note the Chapter title along with the limitations and future works directly beneath.

7.3.1 Chapter 3 - Semi-Automated Estimation of Calories of Meals in Photographs

For future work in determining calories in meals using image processing methods, the data collection process can be extended to more food items to further evaluate this model for attaining calorie content. The system currently has no food type identification techniques employed. The system could incorporate image classification algorithms (Chapter 5 and Chapter 6) to determine food type. Once the food type is identified, the user can use the measurements from the polygonal tool to attain calorie content. Instead of using colour thresholds to identify the fiducial cm^2 marker, other shape detection algorithms could be employed to detect the square, e.g. Hough transform algorithm. Once the square or shape is detected and pixel quantity calculated, the user could use this information to identify food portion size. Coins could also be used a reference point as all currency is set at a particular measurement (according to different international regions) and is much more convenient than using a cm^2 square. These coin measurements could be hard-coded into the application, and the user could then be able to select the currency type for measurement. Edge detection and shape algorithms can then be employed to identify the coin and to use its measurements to calculate food portion area. Future improvements could also include the user taking two images of their meal. The data collection process could be improved by lowering the gram

increments to 5-gram portions instead of 10 grams to help increase accuracy. A photo of the food before it has been eaten and another photo after to pinpoint what has been eaten. This is to increase the accuracy of determining what calories the user has consumed for better dietary management.

7.3.2 Chapter 4 - Using Crowdsourcing for Calorie Adjustment for Image Food Logging

The methods highlighted in Chapter 4 present a novel way in which calorie estimation can be adjusted to promote accurate food logging by using statistical metrics generated through crowdsourcing. The aim of Chapter 4 was to investigate the feasibility of using experts and non-experts to determine calorie content in meals. Chapter presented methods that investigated how 'collective wisdom' can be used to adjust calorie estimations, however, more work needs to be done in this area to investigate how crowdsourcing can be fully utilised to promote dietary management. We have highlighted several areas that could be addressed in future work; (1) extend the survey to allow more participants to complete the survey to gather more estimations for analysis. Extending the survey would allow for greater representation in examining the relationship between both groups; expert and non-expert. (2) The meal images used in this work contain some food items in one image, future work would extend the type of images and include individual food items i.e. a slice of bread, single pieces of fruit. Research has been completed [39] that suggests that individuals can estimate calorie of smaller food meals with higher accuracy compared to larger meals. With this knowledge, future work would include single food item images and smaller meal images to allow users to estimate calorie content and for analysis and to examine the estimation performance accuracy between meal images and food item images (3) future analysis will include methods to reduce further bias between calorie differences of meal images, e.g. a number of meal images type will not be included in the training set analysis but to be used for

testing, e.g. estimations for meal images 1-10 will be used for training and the estimations for meal image type 11-15 will be used to test the calorie difference reduction value generated by the training dataset to see if there is improvement.

7.3.3 Chapter 5 - Feature Fusion for Food Image Classification

Several limitations have been identified in this work. Some of the images used in this work for each category include other objects, or other food items are present in the image. Specific 'non-food' features may be selected and used in the training of the machine learning process, which can result in some misclassifications. Future work will be to address this issue by creating an image dataset using food images that focus in on the food item and texture directly and to ensure that no other non-food items or other food items are present in the scene. In future work, the food items would be segmented from the image and then feature types would be extracted from the segmented image. This would improve the algorithm's accuracy by allowing relevant interest points to be selected and used for training. A number of datasets were used in Chapter 5, for Food-30 the number of images in this work was 100 per category, which can be considered to be a low number in comparison to other works. Future work would address this issue by increasing the number of images in each category and ensure that these images do not contain any other food or non-food items. Other machine learning models would also be considered in future work; further analysis could be undertaken by changing different parameters for each model used, e.g. changing the number of layers in neural network structure along with the number of neurons or changing kernels used in SMO classifier. Other machine learnings could be applied to the image dataset such as Self-Organizing Maps (SOM) or utilise other multi-class classifier approaches and document the performance of these techniques for comparison. For feature extraction, other feature types could be used such as Gabor Filters to extract textual information from the dataset. Research would also focus on developing a hierarchical classification approach to classify

high-level food type using a classifier and then use another classifier dedicate to further classify specific food type.

7.3.4 Chapter 6 - Deep Residual Network Features to Classify Diverse Food Image Datasets

There are some limitations associated with the work presented in Chapter 6 which could be addressed in future works, for example, a comprehensive dataset could be developed under a controlled environment that is representative of a broad range of food items. This dataset could be used with the methods outlined in this work and compared with similar works. This would give a clear indication of the actual performance of using deep feature extraction with machine learning algorithms. Also, a comprehensive study could be completed by fine-tuning a range of CNNs on food datasets and comparing performance using the same pretrained CNN models for deep feature extraction. Further experiments can also be completed by comparing deep features extracted from different layers within a CNN architecture to find what layer is more suitable for generalising between different food classes. In regards to overfitting, particularly for Food-101, future works could include using 10-fold class validation instead of using a 75:25 train/testing split. This would give a more precise indication of the performance of using deep features from ResNet-152 and GoogleNet. Some of the experiments in this work achieved high accuracies, especially for Food/Non-Food classification experiments, however, it is important to note that the number of images contained in Food-5K is relatively small in comparison to other datasets, e.g. Food-11 or Food-101. Further experiments need to be completed in detecting food/non-food in larger food image datasets in using off the shelf deep features.

For RawFoot-DB we used the training and test split provided by authors in [259], however, the authors of RawFooT-DB created subsets of each category, which were based on

lighting condition type. In this work, the aim was to classify food textures across different lighting conditions, however, in future work, we would follow the same procedures and use ResNet-152 features for further comparison. The work presented in Chapter 6 using RawFooT DB used only the 'no variance' dataset in which all lighting conditions are collated together and used for training and testing. In addition authors of Food-101 [213] allocated a testing split that contained images that contained little noise and representative of each class, however in our work using Food-101 extracted features were shuffled using random seed '1' and random state '1' to determine the classification performance of ResNet-152 features when used with images with high level of noise. In future works, we will further evaluate ResNet-152 features following the partition procedure described in [213].

Future work could incorporate hierarchical classification using pretrained CNN features in which a classifier will be used to determine food and non-food images, another classifier will be appended that determines major food groups, and finally, a further classifier will be used after to determine low-level food item. Further experiments with the parameters of machine learning models could also be changed to determine the optimal parameter settings to achieve a high classification accuracy. The presence of noise in the food image datasets may also affect the accuracy, to mitigate these issues, a semi-automated approach could be adopted by using a polygonal tool to draw around the food portion and to ultimately segment the food item. Classification models could then classify the segmented food portion to promote accuracy. For future evaluation, we would also input random noise as feature vectors for trained classifiers to determine food classes and analyse the output and performance. The use of machine learning models using pretrained CNN deep features also has the potential of being used in mobile health solutions. The research presented in Chapter 5 indicate that food logging can be automated using deep features extracted from residual CNNs for high food classification accuracy. From this research presented in Chapter 6, it is clear

that ResNet-152 deep features can distinguish between high-level food categories such as Food/Non-food and echoes other related research in this area. In comparison with other works, ResNet-152 deep features outperform other CNN deep features such as GoogleNet in distinguishing between fine-grained food texture classes in RawFooT-DB and is comparable to other related works [259]. ResNet-152 features encountered some difficulty in classifying Food-101 classes. However, this may be due to the images containing noise in the form of high colour intensities and multiple foods in the same image. However, a reasonable accuracy of 64.98% was achieved. In Food-11 food group classification, deep GoogleNet features were able to achieve high accuracy result when compared to research presented in [209] which used a fine-tuned GoogleNet, and results suggest that a combination of conventional machine learning classifiers combined with CNN features can outperform fine-tuned models for certain image datasets.

Further research would be explored in using deep features extracted from CNN for fine-grained food classification using image types of similar types photograph in real-world environments. Classifying food images in real world environments have been explored in this thesis (Chapter 5, and 6), however further work can be completed in developing an image dataset of different categories of similar food items, e.g. soup dishes and a dataset for salad dishes etc. Deep features would then be extracted from these datasets using a pretrained CNN ResNet-152 model and these features would then be used to train supervised machine learning classifiers. These experiments would evaluate the performance of using deep ResNet-152 features in capturing relevant features that could be used to generalise between low in-between variance in similar dishes, i.e. soup dishes, salad dishes, or meat dishes.

7.4 Conclusion

This PhD thesis provides a series of experiments to inform the development of a framework that combines computer vision and statistical approaches for food image logging and dietary management. Chronic conditions related to obesity and being overweight such as Type-2 diabetes, sleep apnoea, and some cancers are increasing, and food logging has been shown to be an essential method in regards to weight management, as discussed in Chapter 2 Literature Review. Automated food logging is a relatively new approach to dietary management and combines various approaches to predict food items in photographs to determine nutritional content. This technique removes much of the complexity of conventional food logging techniques and promotes usability as the user only needs to photograph food item to document nutritional intake. The research completed in this thesis proposes a dietary management framework that combines computer vision and statistical approaches for dietary management. Chapter 5 and 6 of this thesis demonstrate how computer vision feature extraction approaches can be employed to classify food portions in images in a variety of food image datasets for the following areas; food detection, food texture classification, and food classification in free-living environments. Conventional feature extraction fusion was used across several food image datasets to determine optimal feature combination and applied to machine learning classifiers. Results show that a feature fusion approach is needed in classifying food images in a free-living environment. Deep learning was also investigated in classifying food image datasets for automated food logging discussed in Chapter 6. Results from Chapter 6 show that CNN deep feature extraction, when applied to conventional machine learning classifiers, achieves high accuracy when applied to various food classification areas, i.e. food detection, food group, and specific food item classification. Results show that deep ResNet-152 features can detect food items in images with high accuracy and can generalise efficiently between food groups. Results also suggest that deep residual

features extracted from deep in CNN architecture can achieve high accuracy in classifying isolated food texture patches under different illuminations. However, deep features perform moderately when used to classify images captured in free-living environments which may suggest that an image segmentation or semi-automation approach may be needed to isolate food portion for classification.

Chapter 3 and 4 present novel approaches for calorie estimation and results from Chapter 3 suggest that surface area of certain food item types is a characteristic that can be used to determine nutritional content accurately. The combination of colour segmentation and manual user segmentation can be accurately pinpoint calorie content. Chapter 4 investigates the use of crowdsourcing and results show that there is potential in utilising a crowd of non-experts in accurately determining calorie in food images. A calorie adjustment approach was proposed that utilised crowdsourcing descriptive statistics and crowdsourcing calorie overestimations to develop a rule-based algorithm to adjust calorie predictions to promote accuracy. The dietary management framework combines elements from each study (Figure 7.3) to propose a novel automated food logging system. Potential studies have also been proposed for each Chapter. For Chapter 3, further research with other food types using the approaches proposed in Chapter 3 could be explored. In Chapter 4, further research could combine the techniques used in Chapter 3 to manually segment food portions to enhance food classification accuracy. Methods discussed in Chapter 4 could be used to develop a hierarchical food image classification to determine food group and specific food item. In Chapter 6, further work could be explored by extracting deep features using different layers in a pretrained ResNet-152. Fine-tuning could also be explored in fine-tuning a ResNet-152 model for food image classification and to compare with results achieved in Chapter 6. For Chapter 4, further research could include extending the survey would allow for greater representation in examining the relationship between both groups; expert and non-expert. (2) The meal images used in this work contain some food items in one image, future work

would extend the type of images and include individual food items, i.e. slice of bread, single pieces of fruit. Future work would include single food item images and smaller meal images to allow users to estimate calorie content and for analysis and to examine the estimation performance accuracy between meal images and food item images. Future analysis will include methods to reduce further bias between calorie differences of meal images for calorie adjustment process, e.g. a number of meal images type will not be included in the training set analysis and they will be used for testing, e.g. estimations for meal images 1-10 will be used for training and the estimations for meal image type 11-15 will be used to test the calorie difference reduction value generated by the training dataset to see if there is improvement.

The studies presented in this thesis seek to inform the development of an automated food logging platform that combines novel techniques such as deep learning and crowdsourcing. The use of smartphones has increased dramatically, and users can capture high-quality images using a smartphone camera. Users have the opportunity to document their nutritional intake through an image food logging approach, and research has indicated that using an image-based approach for food logging can promote convenience and usability (in comparison to traditional food logging) [36]. The use of a camera can also be used to ascertain detail information regarding food intake as previous research has used fiducial markers to determine portion size with promising results. The studies presented in this thesis discuss various approaches that could be used for food image classification and calorie calculation and how they could be combined to achieve state-of-the-art automatic food image logging.

7.5 Publications

The following is a list of publications from research presented in this PhD thesis. Each Chapter presents methods and results related, which are also documented in each published work.;

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, “Combining deep residual network features with supervised machine learning algorithms to classify diverse food image datasets,” Comput. Biol. Med., Feb. 2018., DOI:10.1016/j.compbiomed.2018.02.008., ISSN: 00104825. [Journal contribution]

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, “Semi-automated system for predicting calories in photographs of meals,” in 2015 IEEE International Conference on Engineering, Technology and Innovation/ International Technology Management Conference, ICE/ITMC 2015, 2016. [International conference]

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, “A semi-automated food voting classification system: Combining user interaction and Support Vector Machines,” in International Symposium on Technology and Society, Proceedings, 2016, vol. 2016–March. [International conference]

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, “Towards personalised training of machine learning algorithms for food image classification using a smartphone camera,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016, vol. 10069 LNCS, pp. 178–190. [International conference]

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, “A digital technology framework to optimise the self-management of obesity,” in Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct - UbiComp ’16, 2016, pp. 1126–1131. [Workshop contribution]

- **P. McAllister**, H. Zheng, R. Bond, and A. Moorhead, "Comparison of Machine Learning Algorithms in Classifying Segmented Photographs of Food for Food Logging", Proceeding of CERC 2016 Collaborative European Research Conference Cork Institute

of Technology – Cork, Ireland 23 - 24 September 2016 www.cerc–conference.eu ISSN 2220 – 4164. [Regional conference]

- **P. McAllister**, A. Moorhead, R. Bond and H. Zheng, "Automated adjustment of crowd-sourced calorie estimations for accurate food image logging," 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Kansas City, MO, 2017, pp. 1059-1066. doi: 10.1109/BIBM.2017.8217803 [Workshop contribution]

References

[1] M. Di Cesare, et al., Trends in adult body-mass index in 200 countries from 1975 to 2014: a pooled analysis of 1698 population-based measure- ment studies with 19.2 million participants, Lancet 387 (10026) (2016) 1377–1396.

[2] "Obesity", nhs.uk, 2018. [Online]. Available: https://www.nhs.uk/conditions/obesity/. [Accessed: 22- Feb- 2018].

[3] "Obesity and overweight", World Health Organization, 2018. [Online]. Available: http://www.who.int/mediacentre/factsheets/fs311/en/. [Accessed: 22- Feb- 2018].

[4] "Defining Childhood Obesity | Overweight & Obesity | CDC", Cdc.gov, 2018. [Online]. Available: https://www.cdc.gov/obesity/childhood/defining.html. [Accessed: 22- Feb- 2018].

[5] Ebbeling CB, Pawlak DB, Ludwig DS. Childhood obesity: public-health crisis, common sense cure. The Lancet. 2002;360(9331):473–482

[6] J. Steinberger and S. R. Daniels, “Obesity, Insulin Resistance, Diabetes, and Cardiovascular Risk in Children,” Circulation, vol. 107, no. 10, pp. 1448–1453, 2003.

[7] J. Sorof and S. Daniels, “Obesity hypertension in children: A problem of epidemic proportions,” Hypertension, vol. 40, no. 4. pp. 441–447, 2002.

[8] M. K. Serdula, D. Ivery, R. J. Coates, D. S. Freedman, D. F. Williamson, and T. Byers, "Do obese children become obese adults? A review of the literature.," Preventive medicine, vol. 22, no. 2. pp. 167–77, 1993.

[9] "Health Survey (NI) 2016/17", Health, 2018. [Online]. Available: https://www.health-ni.gov.uk/news/health-survey-ni-201617. [Accessed: 22- Feb- 2018].

[10] House of Commons, 'Obesity Statistics',Briefing Paper', January 2017.

[11] "Health Survey (NI) 2015/16", Health, 2018. [Online]. Available: https://www.health-ni.gov.uk/news/health-survey-ni-201516. [Accessed: 22- Feb- 2018].

[12] "Health survey Northern Ireland: first results | Department of Health", Health, 2018. [Online]. Available: https://www.health-ni.gov.uk/publications/health-survey-northern-ireland-first-results. [Accessed: 22- Feb- 2018].

[13] "Reducing obesity: future choices - GOV.UK", Gov.uk, 2018. [Online]. Available: https://www.gov.uk/government/publications/reducing-obesity-future-choices. [Accessed: 22- Feb- 2018].

[14] S. Mayor, "Over a third of children aged 10-11 in England are overweight or obese," BMJ (Online), vol. 355. 2016.

[15] Mokdad AH, Marks JS, Stroup DF, Gerberding JL. Actual causes of death in the United States, 2000. JAMA 2004;291:1238–45

[16] "Obesity", WHO Western Pacific Region, 2018. [Online].
Available: http://www.wpro.who.int/mediacentre/factsheets/obesity/en/. [Accessed: 22- Feb- 2018].

[17] "House of Commons - Childhood obesity-brave and bold action - Health Committee", Publications.parliament.uk, 2018. [Online].

Available: https://publications.parliament.uk/pa/cm201516/cmselect/cmhealth/465/46502.htm. [Accessed: 22- Feb- 2018].

[18] A. Dee et al., “The direct and indirect costs of both overweight and obesity: A systematic review,” BMC Res. Notes, vol. 7, no. 1, 2014.

[19] "McKinsey: Obesity costs UK society 73 billion per year", Consultancy.uk, 2018. [Online]. Available: https://www.consultancy.uk/news/1278/mckinsey-obesity-costs-uk-society-73-billion-per-year. [Accessed: 22- Feb- 2018].

[20] M. L. Butryn, S. Phelan, J. O. Hill, and R. R. Wing, “Consistent self-monitoring of weight: A key component of successful weight loss maintenance,” Obesity, vol. 15, no. 12, pp. 3091–3096, 2007.

[21] L. E. Burke, J. Wang, and M. A. Sevick, “Self-Monitoring in Weight Loss: A Systematic Review of the Literature,” J. Am. Diet. Assoc., vol. 111, no. 1, pp. 92–102, 2011.

[22] J. P. Chaput, L. Klingenberg, A. Astrup, and A. M. Sjödin, “Modern sedentary activities promote overconsumption of food in our current obesogenic environment,” Obes. Rev., vol. 12, no. 501, 2011.

[23] "Understanding Teens and Their Smartphone Habits - eMarketer", Emarketer.com, 2018. [Online]. Available: https://www.emarketer.com/Article/Understanding-Teens-Their-Smartphone-Habits/1016423. [Accessed: 22- Feb- 2018].

[24] "Adults’ media use and attitudes", Ofcom, 2018. [Online]. Available: https://www.ofcom.org.uk/resear and-data/media-literacy-research/adults/adults-media-use-and-attitudes. [Accessed: 22- Feb- 2018].

[25] J. A. D. Long and K. R. Stevens, “Using technology to promote self-efficacy for healthy eating in adolescents,” J. Nurs. Scholarsh., vol. 36, no. 2, pp. 134–139, 2004.

[26] C. C. Curioni and P. M. Lourenço, "Long-term weight loss after diet and exercise: A systematic review," International Journal of Obesity, vol. 29, no. 10. pp. 1168–1174, 2005.

[27] K. Elfhag and S. Rössner, "Who succeeds in maintaining weight loss? A conceptual review of factors associated with weight loss maintenance and weight regain.," Obes. Rev., vol. 6, no. 1, pp. 67–85, 2005.

[28] M. C. Carter, V. J. Burley, C. Nykjaer, and J. E. Cade, "Adherence to a smartphone application for weight loss compared to website and paper diary: pilot randomized controlled trial.," J. Med. Internet Res., vol. 15, no. 4, 2013.

[29] A. Andrew, G. Borriello, and J. Fogarty, "Simplifying Mobile Phone Food Diaries: Design and Evaluation of a Food Index-Based Nutrition Diary," PervasiveHealth, pp. 260–263, 2013.

[30] R. C. Baker and D. S. Kirschenbaum, "Self-monitoring may be necessary for successful weight control," Behav. Ther., vol. 24, no. 3, pp. 377–394, 1993.

[31] N. D. Peterson, K. R. Middleton, L. M. Nackers, K. E. Medina, V. A. Milsom, and M. G. Perri, "Dietary self-monitoring and long-term success with weight management," Obesity, vol. 22, no. 9, pp. 1962–1967, 2014.

[32] B. Y. Laing et al., "Effectiveness of a smartphone application for weight loss compared with usual care in overweight primary care patients: a randomized, controlled trial.," Ann. Intern. Med., vol. 161, no. 10 Suppl, pp. S5-12, 2014.

[33] M. Duncan et al., "Effectiveness of a web- and mobile phone-based intervention to promote physical activity and healthy eating in middle-Aged males: Randomized controlled trial of the manup study," J. Med. Internet Res., vol. 16, no. 6, 2014.

[34] L. E. Burke et al., “Using mHealth technology to enhance self-monitoring for weight loss: A randomized trial,” Am. J. Prev. Med., vol. 43, no. 1, pp. 20–26, 2012.

[35] G. Eysenbach, “The law of attrition,” Journal of Medical Internet Research, vol. 7, no. 1. 2005.]

[36] M. J. Hutchesson, M. E. Rollo, R. Callister, and C. E. Collins, “Self-Monitoring of Dietary Intake by Young Women: Online Food Records Completed on Computer or Smartphone Are as Accurate as Paper-Based Food Records but More Acceptable,” J. Acad. Nutr. Diet., vol. 115, no. 1, pp. 87–94, 2015.

[37] C. M. Wharton, C. S. Johnston, B. K. Cunningham, and D. Sterner, “Dietary Self-Monitoring, But Not Dietary Quality, Improves With Use of Smartphone App Technology in an 8-Week Weight Loss Trial,” J. Nutr. Educ. Behav., vol. 46, no. 5, pp. 440–444, 2014.

[38] F. Cordeiro et al., “Barriers and Negative Nudges,” in Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI ’15, 2015, pp. 1159–1162.

[39] Wansink and P. Chandon, Meal size, not body size, explains errors in estimating the calorie content of meals, Ann. Intern. Med., vol. 145, no. 5, pp. 326-332, 2006.

[40] J. Noronha, E. Hysen, H. Zhang, and K. Z. Gajos, “Platemate: Crowdsourcing Nutritional Analysis from Food Photographs,” in Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology, 2011, pp. 1–12.

[41] M. Vazquez-Briseno, C. Navarro-Cota, J. I. Nieto- Hipolito, E. Jimenez-Garcia, and J. D. Sanchez-Lopez, “A proposal for using the internet of things concept to increase children’s health awareness,” CONIELECOMP 2012, 22nd Int. Conf. Electr. Commun. Comput., pp. 168–172, Feb. 2012

[42] C. K. Martin, H. Han, S. M. Coulon, H. R. Allen, C. M. Champagne, and S. D. Anton, "A novel method to remotely measure food intake of free-living individuals in real time: The remote food photography method," Br. J. Nutr., vol. 101, no. 3, pp. 446–456, 2009.

[43] A. Naska, E. Valanou, E. Peppa, M. Katsoulis, A. Barbouni, and A. Trichopoulou, "Evaluation of a digital food photography atlas used as portion size measurement aid in dietary surveys in Greece," Public Health Nutrition, vol. 19, no. 13. pp. 2369–2376, 2016.

[44] S. I. Kirkpatrick et al., "The Use of Digital Images in 24-Hour Recalls May Lead to Less Misestimation of Portion Size Compared with Traditional Interviewer-Administered Recalls," J. Nutr., vol. 146, no. 12, pp. 2567–2573, 2016.

[45] E. Valanou, A. Naska, A. Barbouni, M. Katsoulis, E. Peppa, P. Vidalis, and A. Trichopoulou, "Evaluation of food photographs assessing the dietary intake of children up to 10 years old," Public Health Nutrition, pp. 1–8, 2017.

[46] R. Z. Franco, R. Fallaize, J. A. Lovegrove, and F. Hwang, "Popular Nutrition-Related Mobile Apps: A Feature Assessment," JMIR mHealth uHealth, vol. 4, no. 3, p. e85, 2016.

[47] M. L. Ovaskainen et al., "Accuracy in the estimation of food servings against the portions in food photographs," Eur. J. Clin. Nutr., vol. 62, no. 5, pp. 674–681, 2008.

[48] E. Helander, K. Kaipainen, I. Korhonen, and B. Wansink, "Factors related to sustained use of a free mobile app for dietary self-monitoring with photography and peer feedback: Retrospective cohort study," J. Med. Internet Res., vol. 16, no. 4, 2014.

[49] F. Cordeiro, E. Bales, E. Cherry, and J. Fogarty, "Rethinking the Mobile Food Journal: Exploring Opportunities for Lightweight Photo-Based Capture," Proc. 33rd Annu. ACM Conf. Hum. Factors Comput. Syst. - CHI '15, pp. 3207–3216, 2015.

[50] G. O'Loughlin et al., "Using a wearable camera to increase the accuracy of dietary analysis," Am. J. Prev. Med., vol. 44, no. 3, pp. 297–301, 2013.

[51] C.-F. Chung, E. Agapie, J. Schroeder, S. Mishra, J. Fogarty, and S. A. Munson, "When Personal Tracking Becomes Social: Examining the Use of Instagram for Healthy Eating," Proc. SIGCHI Conf. Hum. Factors Comput. Syst. (CHI 2017), pp. 1674–1687, 2017.

[52] J. Pollak et al., "It's Time to Eat! Using Mobile Games to Promote Healthy Eating," IEEE Pervasive Comput., vol. 9, no. 3, pp. 21–27, 2010.

[53] J. M. Vaterlaus, E. V. Patten, C. Roche, and J. A. Young, "Gettinghealthy: The perceived influence of social media on young adult health behaviors," Comput. Human Behav., vol. 45, pp. 151–157, 2015.

[54] R. S. Gruver et al., "A Social Media Peer Group Intervention for Mothers to Prevent Obesity and Promote Healthy Growth from Infancy: Development and Pilot Trial," JMIR Res. Protoc., vol. 5, no. 3, p. e159, 2016.

[55] F. Jimoh et al., "Comparing Diet and Exercise Monitoring Using Smartphone App and Paper Diary: A Two-Phase Intervention Study," JMIR mHealth uHealth, vol. 6, no. 1, p. e17, 2018.

[56] L. Dennison, L. Morrison, G. Conway, and L. Yardley, "Opportunities and challenges for smartphone applications in supporting health behavior change: qualitative study.," J. Med. Internet Res., vol. 15, no. 4, 2013.

[57] A. Direito, L. Pfaeffli Dale, E. Shields, R. Dobson, R. Whittaker, and R. Maddison, "Do physical activity and dietary smartphone applications incorporate evidence-based behaviour change techniques?," BMC Public Health, vol. 14, no. 1, 2014.

[58] S. L. Casperson, J. Sieling, J. Moon, L. Johnson, J. N. Roemmich, and L. Whigham, "A Mobile Phone Food Record App to Digitally Capture Dietary Intake for Adolescents in a Free-Living Environment: Usability Study," JMIR mHealth uHealth, vol. 3, no. 1, p. e30, 2015.

[59] C. K. Martin, T. Nicklas, B. Gunturk, J. B. Correa, H. R. Allen, and C. Champagne, "Measuring food intake with digital photography," J. Hum. Nutr. Diet., vol. 27, no. SUPPL.1, pp. 72–81, 2014.

[60] N. Amougou et al., "Development and validation of two food portion photograph books to assess dietary intake among adults and children in Central Africa," Br. J. Nutr., vol. 115, no. 5, pp. 895–902, 2016.

[61] M. Bouchoucha et al., "Development and validation of a food photography manual, as a tool for estimation of food portion size in epidemiological dietary surveys in Tunisia," Libyan J. Med., vol. 11, 2016.

[62] C. K. Martin et al., "Validity of the remote food photography method (RFPM) for estimating energy and nutrient intake in near real-time," Obesity, vol. 20, no. 4, pp. 891–899, 2012.

[63] C. K. Martin, S. Kaya, and B. K. Gunturk, "Quantification of food intake using food image analysis," in Proceedings of the 31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society: Engineering the Future of Biomedicine, EMBC 2009, 2009, pp. 6869–6872.

[64] A. D. Altazan et al., "Development and application of the remote food photography method to measure food intake in exclusively milk fed infants: A laboratory-based study," PLoS One, vol. 11, no. 9, 2016.

[65] A. D. Altazan et al., "Development and application of the remote food photography method to measure food intake in exclusively milk fed infants: A laboratory-based study," PLoS One, vol. 11, no. 9, 2016.

[66] A. D. Lassen, S. Poulsen, L. Ernst, K. K. Andersen, A. Biltoft-Jensen, and I. Tetens, "Evaluation of a digital method to assess evening meal intake in a free-living adult population," Food Nutr. Res., vol. 54, 2010.

[67] Á. McConnon et al., "The Internet for weight control in an obese sample: Results of a randomised controlled trial," BMC Health Serv. Res., vol. 7, 2007.

[68] L. G. Womble, T. A. Wadden, B. G. McGuckin, S. L. Sargent, R. A. Rothman, and E. S. Krauthamer-Ewing, "A randomized controlled trial of a commercial internet weight loss program," Obes. Res., vol. 12, no. 6, pp. 1011–1018, 2004.

[69] M. Rabbi, A. Pfammatter, M. Zhang, B. Spring, and T. Choudhury, "Automated Personalized Feedback for Physical Activity and Dietary Behavior Change With Mobile Phones: A Randomized Controlled Trial on Adults," JMIR mHealth uHealth, vol. 3, no. 2, p. e42, 2015.

[70] H. Forster, M. C. Walsh, M. J. Gibney, L. Brennan, and E. R. Gibney, "Personalised nutrition: The role of new dietary assessment methods," in Proceedings of the Nutrition Society, 2016, vol. 75, no. 1, pp. 96–105.

[71] R. Laganière, OpenCV 2 Computer Vision Application Programming Cookbook. 2011.

[72] A. Géron, Hands-on machine learning with Scikit-Learn and TensorFlow. Beijing [etc.]: O'Reilly, 2017.

[73] D. G. Lowe, "Object recognition from local scale-invariant features," in Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999, pp. 1150–1157 vol.2.

[74] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-Up Robust Features (SURF)," Comput. Vis. Image Underst., vol. 110, no. 3, pp. 346–359, 2008.

[75] R. Chincha and Y. L. Tian, "Finding objects for blind people based on SURF features," in 2011 IEEE International Conference on Bioinformatics and Biomedicine Workshops, BIBMW 2011, 2011, pp. 526–527.

[76] S. Govindaraju and G. P. Ramesh Kumar, "A novel content based medical image retrieval using SURF features," Indian J. Sci. Technol., vol. 9, no. 20, 2016.

[77] P. Pouladzadeh, S. Shirmohammadi, and T. Arici, "Intelligent SVM based food intake measurement system," in 2013 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications, CIVEMSA 2013 - Proceedings, 2013, pp. 87–92.

[78] R. S. Hunter, "Photoelectric Color Difference Meter," J. Opt. Soc. Am., vol. 48, no. 12, p. 985, 1958.

[79] D. Ravi, B. Lo, and G. Z. Yang, "Real-time food intake classification and energy expenditure estimation on a mobile device," in 2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks, BSN 2015, 2015.

[80] M. Tkalčič and J. F. Tasič, "Colour spaces - Perceptual, historical and applicational background," in IEEE Region 8 EUROCON 2003: Computer as a Tool - Proceedings, 2003, vol. A, pp. 304–308.

[81] S. E. A. Raza, G. Prince, J. P. Clarkson, and N. M. Rajpoot, “Automatic detection of diseased tomato plants using thermal and stereo visible light images,” PLoS One, vol. 10, no. 4, 2015.

[82] S. L. Phung, A. Bouzerdoum, and D. Chai, “Skin segmentation using color pixel classification: Analysis and comparison,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 1, pp. 148–154, 2005.

[83] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk, “Frequency-tuned salient region detection,” in 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 1597–1604.

[84] R. Cucchiara, C. Grana, M. Piccardi, a. Prati, and S. Sirotti, “Improving shadow suppression in moving object detection with HSV color information,” ITSC 2001. 2001 IEEE Intell. Transp. Syst. Proc. (Cat. No.01TH8585), pp. 334–339, 2001.

[85] Rein-Lien Hsu, M. Abdel-Mottaleb, and A. K. Jain, “Face detection in color images,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 5, pp. 696–706, 2002.

[86] K. Van De Sande, T. Gevers, and C. Snoek, “Evaluating color descriptors for object and scene recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 32, no. 9, pp. 1582–1596, 2010.

[87] W. Chen, Y. Q. Shi, and G. Xuan, “Identifying Computer Graphics using HSV Color Model and Statistical Moments of Characteristic Functions,” Multimed. Expo, 2007 IEEE Int. Conf., pp. 1123–1126, 2007.

[88] S. Soleimanizadeh, D. Mohamad, T. Saba, and A. Rehman, “Recognition of Partially Occluded Objects Based on the Three Different Color Spaces (RGB, YCbCr, HSV),” 3D Res., vol. 6, no. 3, 2015.

[89] H. D. Cheng, X. H. Jiang, Y. Sun, and J. Wang, "Color image segmentation: Advances and prospects," Pattern Recognit., vol. 34, no. 12, pp. 2259–2281, 2001.

[90] Q. Zhu, S. Avidan, M. C. Yeh, and K. T. Cheng, "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients," IEEE Conf. Comput. Vis. Pattern Recognit., vol. 2, pp. 1491–1498, 2006.

[91] Y. Yamauchi, C. Matsushima, T. Yamashita, and H. Fujiyoshi, "Relational HOG feature with wild-card for object detection," in Proceedings of the IEEE International Conference on Computer Vision, 2011, pp. 1785–1792.

[92] M. Dahmane and J. Meunier, "Emotion recognition using dynamic grid-based HoG features," in 2011 IEEE International Conference on Automatic Face and Gesture Recognition and Workshops, FG 2011, 2011, pp. 884–888.

[93] C. Gottschlich, E. Marasco, A. Y. Yang, and B. Cukic, "Fingerprint liveness detection based on histograms of invariant gradients," in IJCB 2014 - 2014 IEEE/IAPR International Joint Conference on Biometrics, 2014.

[94] R. M. Haralick, "Statistical and structural approaches to texture," Proc. IEEE, vol. 67, no. 5, pp. 786–804, 1979.

[95] A. F. Costa, G. Humpire-Mamani, and A. J. M. Traina, "An efficient algorithm for fractal analysis of textures," in Brazilian Symposium of Computer Graphic and Image Processing, 2012, pp. 39–46.

[96] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 7, pp. 971–987, 2002.

[97] T. Ahonen, A. Hadid, and M. Pietikäinen, “Face description with local binary patterns: Application to face recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 28, no. 12, pp. 2037–2041, 2006.

[98] T. Huynh, R. Min, and J. L. Dugelay, “An efficient LBP-based descriptor for facial depth images applied to gender recognition using RGB-D face data,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2013, vol. 7728 LNCS, no. PART 1, pp. 133–145.

[99] C. Shan, S. Gong, and P. W. McOwan, “Facial expression recognition based on Local Binary Patterns: A comprehensive study,” Image Vis. Comput., vol. 27, no. 6, pp. 803–816, 2009.

[100] "Google Summer of Code: patent-free Face Detection for Scikit-image in Python. Introduction", Warmspringwinds.github.io, 2018. [Online]. Available: https://goo.gl/Djk9fB. [Accessed: 08- Mar- 2018].

[101] J. R. Movellan, “Tutorial on Gabor Filters,” Response, vol. 49, pp. 1–23, 2002.

[102] K. Murthy, "Gabor Filters : A Practical Overview", Computer Vision Tutorials, 2018. [Online]. Available: https://cvtuts.wordpress.com/2014/04/27/gabor-filters-a-practical-overview/. [Accessed: 08- Mar- 2018].

[103] J. Yang, L. Liu, T. Jiang, and Y. Fan, “A modified Gabor filter design method for fingerprint image enhancement,” Pattern Recognit. Lett., vol. 24, no. 12, pp. 1805–1817, 2003.

[104] R. Baran, P. Partila, and R. Wilk, “Automated Text Detection and Character Recognition in Natural Scenes Based on Local Image Features and Contour Processing Techniques,” in Intelligent Human Systems Integration, 2018, pp. 42–48.

[105] S. Logesh Kumar, M. Swathy, S. Sathish, J. Sivaraman, and M. Rajasekar, "Identification of lung cancer cell using watershed segmentation on CT images," Indian J. Sci. Technol., vol. 9, no. 1, 2016.

[106] L. K. Soh and C. Tsatsoulis, "Texture analysis of sar sea ice imagery using gray level co-occurrence matrices," IEEE Trans. Geosci. Remote Sens., vol. 37, no. 2 I, pp. 780–795, 1999.

[107] A. Mohd. Khuzi, R. Besar, W. M. D. Wan Zaki, and N. N. Ahmad, "Identification of masses in digital mammogram using gray level co-occurrence matrices," Biomed. Imaging Interv. J., vol. 5, no. 3, 2009.

[108] S. Beura, B. Majhi, and R. Dash, "Mammogram classification using two dimensional discrete wavelet transform and gray-level co-occurrence matrix for detection of breast cancer," Neurocomputing, vol. 154, pp. 1–14, 2015.

[109] M. Partio, B. Cramariuc, M. Gabbouj, and A. Visa, "Rock texture retrieval using gray level co-occurrence matrix," Proc. 5th Nord. Signal . . . , 2002.

[110] R. Maini and H. Aggarwal, "Study and comparison of various image edge detection techniques," Int. J. Image Process., vol. 3, no. 1, pp. 1–11, 2009.

[111] S. Tabbone and D. Ziou, "Edge Detection Techniques - An Overview," Int. J. Pattern Recognit. Image Anal., vol. 8, pp. 537–559, 1998.

[112] N. Kanopoulos, N. Vasanthavada, and R. L. Baker, "Design of an Image Edge Detection Filter Using the Sobel Operator," IEEE J. Solid-State Circuits, vol. 23, no. 2, pp. 358–367, 1988.

[113] C. Wählby, I. M. Sintorn, F. Erlandsson, G. Borgefors, and E. Bengtsson, "Combining intensity, edge and shape information for 2D and 3D segmentation of cell nuclei in tissue sections," J. Microsc., vol. 215, no. 1, pp. 67–76, 2004.

[114] S. Jabri, Z. Duric, H. Wechsler, and a. Rosenfeld, “Detection and location of people in video images using adaptive fusion of color and edge information,” Proc. 15th Int. Conf. Pattern Recognition. ICPR-2000, vol. 4, pp. 627–630, 2000.

[115] J. Canny, “A Computational Approach to Edge Detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. PAMI-8, no. 6, pp. 679–698, 1986.

[116] "OpenCV: Canny Edge Detection", Docs.opencv.org, 2018. [Online]. Available: https://docs.opencv.org/3.0.0/da/d22/tutorial_py_canny.html. [Accessed: 08- Mar- 2018].

[117] J. Cheng, R. Xue, W. Lu, and R. Jia, “Segmentation of medical images with Canny operator and GVF Snake model,” in Proceedings of the World Congress on Intelligent Control and Automation (WCICA), 2008, pp. 1777–1780.

[118] C. Rother, V. Kolmogorov, and A. Blake, “‘GrabCut’: interactive foreground extraction using iterated graph cuts,” ACM Trans. Graph., vol. 23, no. 3, p. 309, 2004.

[119] A. Hernández, M. Reyes, S. Escalera, and P. Radeva, “Spatio-Temporal Grabcut human segmentation for face and pose recovery,” in 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops, CVPRW 2010, 2010, pp. 33–40.

[120] A. Papazoglou and V. Ferrari, “Fast object segmentation in unconstrained video,” in Proceedings of the IEEE International Conference on Computer Vision, 2013, pp. 1777–1784.

[121] Y. y. Liu, S. y. Zhou and J. h. Sun, "Detection of Ginseng leaf cicatrices base on K-means clustering algorithm," 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Shanghai, China, 2017, pp. 1-5. doi: 10.1109/CISP-BMEI.2017.8302051

[122] K. Yadanar Win, S. Choomchuay and K. Hamamoto, "K mean clustering based automated segmentation of overlapping cell nuclei in pleural effusion cytology images," 2017 International Conference on Advanced Technologies for Communications (ATC), Quy Nhon, 2017, pp. 265-269. doi: 10.1109/ATC.2017.8167630

[123] Jidong Lv, Genrong Shen and Zhenghua Ma, "Acquisition of fruit region in green apple image based on the combination of segmented regions," 2017 2nd International Conference on Image, Vision and Computing (ICIVC), Chengdu, 2017, pp. 332-335. doi: 10.1109/ICIVC.2017.7984572

[124] I.H. Witten, E. Frank, M. a Hall, Data Mining: Practical Machine Learning Tools and Techniques, 2011.

[125] P. Pouladzadeh, G. Villalobos, R. Almaghrabi, and S. Shirmohammadi, “A novel SVM based food recognition method for calorie measurement applications,” in Proceedings of the 2012 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2012, 2012, pp. 495–498.

[126] T. Malisiewicz, A. Gupta, A.A. Efros, Ensemble of exemplar-SVMs for object detection and beyond, In: Proceedings of the IEEE International Conference on Com- puter Vision, 2011, pp. 89–96

[127] E. Osuna, R. Freund, and F. Girosit, “Training support vector machines: an application to face detection,” Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., pp. 130–136, 1997.

[128] C.-W. Hsu, C.-J. Lin, A comparison of methods for multiclass sup- port vector machines, IEEE Trans. Neural Network. 13 (2) (2002) 415–425.

[129] J. C. Platt, “Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines,” 1998.

[130] T. Li, C. Zhang, and M. Ogihara, "A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression," Bioinformatics, vol. 20, no. 15, pp. 2429–2437, 2004.

[131] Java (Convolutional or Fully-connected) Neural Network Implementation, GitHub, 2017, [Online]. Available: https://github.com/amten/NeuralNetwork/releases/ tag/v1.1, [Accessed: 18- Sep- 2017].

[132] S. R. Safavian and D. Landgrebe, "A Survey of Decision Tree Classifier Methodology A SURVEY OF DECISION TREE CLASSIFIER METHODOLOGY 1," Electr. Eng., vol. 21, no. 3, pp. 660–674, 1991.

[133] L. Breiman, "Random forest," Mach. Learn., vol. 45, no. 5, pp. 1–35, 1999.

[134] H. Zhang, The optimality of naive bayes, Proc. Seventeenth Int. Florida Artif. In- tell. Res. Soc. Conf. FLAIRS 2004 1 (2) (2004) 16.

[135] N. Ravi, N. Dandekar, P. Mysore, and M. Littman, "Activity recognition from accelerometer data," Proc. Natl. . . . , pp. 1541–1546, 2005.

[136] K. Aizawa, Y. Maruyama, H. Li, C. Morikawa, and G. C. De Silva, "Food balance estimation by using personal dietary tendencies in a multimedia food log," IEEE Trans. Multimed., vol. 15, no. 8, pp. 2176–2185, 2013.

[137] J. Chen, H. Huang, S. Tian, and Y. Qu, "Feature selection for text classification with Naïve Bayes," Expert Syst. Appl., vol. 36, no. 3 PART 1, pp. 5432–5435, 2009.

[138] N. Situ, X. Yuan, J. Chen, and G. Zouridakis, "Malignant melanoma detection by Bag-of-Features classification.," Conf. Proc. ... Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. IEEE Eng. Med. Biol. Soc. Annu. Conf., vol. 2008, pp. 3110–3, 2008.

[139] Y.-G. Jiang, C.-W. Ngo, and J. Yang, “Towards optimal bag-of-features for object categorization and semantic video retrieval,” Proc. 6th ACM Int. Conf. Image video Retr. - CIVR ’07, pp. 494–501, 2007.

[140] D. S. Pérez, F. Bromberg, and C. A. Diaz, “Image classification for detection of winter grapevine buds in natural conditions using scale-invariant features transform, bag of features and support vector machines,” Comput. Electron. Agric., vol. 135, pp. 81–95, 2017.

[141] M. M. Anthimopoulos, L. Gianola, L. Scarnato, P. Diem, and S. G. Mougiakakou, “A food recognition system for diabetic patients based on an optimized bag-of-features model,” IEEE J. Biomed. Heal. Informatics, vol. 18, no. 4, pp. 1261–1271, 2014.

[142] S. Giovany, A. Putra, A. S. Hariawan, and L. A. Wulandhari, “Machine Learning and SIFT Approach for Indonesian Food Image Recognition,” in Procedia Computer Science, 2017, vol. 116, pp. 612–620.

[143] M. N. B. Razali, N. Manshor, A. A. Halin, N. Mustapha, and R. Yaakob, “Analysis of SURF and SIFT Representations to recognize food objects,” J. Telecommun. Electron. Comput. Eng., vol. 9, no. 2–12, pp. 81–88, 2017.

[144] P. McAllister, H. Zheng, R. Bond and A. Moorhead, "A semi-automated food voting classification system: Combining user interaction and Support Vector Machines," 2015 IEEE International Symposium on Technology and Society (ISTAS), Dublin, 2015, pp. 1-7. doi: 10.1109/ISTAS.2015.7439433

[145] Y. He, C. Xu, N. Khanna, C. J. Boushey, and E. J. Delp, “Analysis of food images: Features and classification,” in 2014 IEEE International Conference on Image Processing, ICIP 2014, 2014, pp. 2744–2748.

[146] Y. Matsuda, H. Hoashi, and K. Yanai, “Recognition of multiple-food images by detecting candidate regions,” in Proceedings - IEEE International Conference on Multimedia and Expo, 2012, pp. 25–30.

[147] W. Wang, P. Duan, W. Zhang, F. Gong, P. Zhang and Y. Rao, "Towards a Pervasive Cloud Computing Based Food Image Recognition," 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing, Beijing, 2013, pp. 2243-2244. doi: 10.1109/GreenCom-iThings-CPSCom.2013.425

[148] M. Chen, K. Dhingra, W. Wu, L. Yang, R. Sukthankar, and J. Yang, “PFID: Pittsburgh Fast-food Image Dataset,” in Proceedings - International Conference on Image Processing, ICIP, 2009, pp. 289–292.

[149] G. M. Farinella, M. Moltisanti, and S. Battiato, “Classifying food images represented as Bag of Textons,” in 2014 IEEE International Conference on Image Processing, ICIP 2014, 2014, pp. 5212–5216.

[150] Y. Kawano and K. Yanai, “FoodCam: A real-time food recognition system on a smartphone,” Multimed. Tools Appl., vol. 74, no. 14, pp. 5263–5287, 2015.

[151] D. Ravi, B. Lo, and G. Z. Yang, “Real-time food intake classification and energy expenditure estimation on a mobile device,” in 2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks, BSN 2015, 2015.

[152] P. Pouladzadeh, S. Shirmohammadi, and R. Al-Maghrabi, “Measuring calorie and nutrition from food image,” IEEE Trans. Instrum. Meas., vol. 63, no. 8, pp. 1947–1956, 2014.

[153] S. Sasano, X. H. Han, and Y. W. Chen, “Food recognition by combined bags of color features and texture features,” in Proceedings - 2016 9th International Congress on

Image and Signal Processing, BioMedical Engineering and Informatics, CISP-BMEI 2016, 2017, pp. 815–819.

[154] S. Lu, Z. Lu, P. Phillips, S. Wang, J. Wu, and Y. Zhang, “Fruit classification by HPA-SLFN,” in 2016 8th International Conference on Wireless Communications and Signal Processing, WCSP 2016, 2016.

[155] Z. Zong, D. T. Nguyen, P. Ogunbona, and W. Li, “On the combination of local texture and global structure for food classification,” in Proceedings - 2010 IEEE International Symposium on Multimedia, ISM 2010, 2010, pp. 204–211.

[156] M. Anthimopoulos, J. Dehais, P. Diem, and S. Mougiakakou, “Segmentation and recognition of multi-food meal images for carbohydrate counting,” in 13th IEEE International Conference on BioInformatics and BioEngineering, IEEE BIBE 2013, 2013.

[157] S. Lu, Z. Lu, P. Phillips, S. Wang, J. Wu and Y. Zhang, "Fruit classification by HPA-SLFN," 2016 8th International Conference on Wireless Communications & Signal Processing (WCSP), Yangzhou, 2016, pp. 1-5. doi: 10.1109/WCSP.2016.7752639

[158] Q. Chen and E. Agu, “Exploring Statistical GLCM Texture Features for Classifying Food Images,” in Proceedings - 2015 IEEE International Conference on Healthcare Informatics, ICHI 2015, 2015, p. 453.

[159] Basavaraj .S, Anami and Vishwanath, C.Burkpalli, “Texture based Identification and Classification of Bulk Sugary Food Objects", ICGST-GVIP Journal, ISSN: 1687-398X, Volume 9, Issue 4, 2009.

[160] T. Negri et al., “A robust descriptor for color texture classification under varying illumination,” in Proc. of the 12th Int. Joint Conf. on Computer Vision, Imaging and

Computer Graphics Theory and Applications (VISAPP), Porto, Portugal, pp. 378–388 (2017).

[161] Y. Zhang, S. Wang, G. Ji, and P. Phillips, "Fruit classification using computer vision and feedforward neural network," J. Food Eng., vol. 143, pp. 167–177, 2014.

[162] M.-Y. Chen et al., "Automatic Chinese food identification and quantity estimation," SIGGRAPH Asia, vol. 1, no. 212, pp. 1–4, 2012.

[163] Y. Kawano and K. Yanai, "Food image recognition with deep convolutional features," in Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct Publication - UbiComp '14 Adjunct, 2014, pp. 589–593.

[164] W. Zhang, Q. Yu, B. Siddiquie, A. Divakaran, and H. Sawhney, "'Snap-n-Eat': Food recognition and nutrition estimation on a smartphone," J. Diabetes Sci. Technol., vol. 9, no. 3, pp. 525–533, 2015.

[165] H. Hoashi, T. Joutou, and K. Yanai, "Image recognition of 85 food categories by feature fusion," in Proceedings - 2010 IEEE International Symposium on Multimedia, ISM 2010, 2010, pp. 296–301.

[166] Y. Kawano and K. Yanai, "FoodCam: A real-time mobile food recognition system employing Fisher Vector," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2014, vol. 8326 LNCS, no. PART 2, pp. 369–373.

[167] C. Liu, Y. Cao, Y. Luo, G. Chen, V. Vokkarane, and Y. Ma, "Deepfood: Deep learning-based food image recognition for computer-aided dietary assessment," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016, vol. 9677, pp. 37–48.

[168] K. Yanai, R. Tanno, and K. Okamoto, "Efficient Mobile Implementation of A CNN-based Object Recognition System," in Proceedings of the 2016 ACM on Multimedia Conference - MM '16, 2016, pp. 362–366.

[169] S. Yang, M. Chen, D. Pomerleau, and R. Sukthankar, "Food recognition using statistics of pairwise local features," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010, pp. 2249–2256.

[170] Y. Kawano and K. Yanai, "Real-Time Mobile Food Recognition System," Comput. Vis. Pattern Recognit. Work. (CVPRW), 2013 IEEE Conf., pp. 1–7, 2013.

[171] O. Beijbom, N. Joshi, D. Morris, S. Saponas, and S. Khullar, "Menu-match: Restaurant-specific food logging from images," in Proceedings - 2015 IEEE Winter Conference on Applications of Computer Vision, WACV 2015, 2015, pp. 844–851.

[172] S. Inunganbi, A. Seal, and P. Khanna, "Classification of Food Images through Interactive Image Segmentation," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2018, vol. 10752 LNAI, pp. 519–528.

[173] W. Wen and Y. Jie, "Fast food recognition from videos of eating for calorie estimation," in Proceedings - 2009 IEEE International Conference on Multimedia and Expo, ICME 2009, 2009, pp. 1210–1213.

[174] R. R. Hariadi, W. N. Khotimah, and E. A. Wiyono, "Design and implementation of food nutrition information system using SURF and FatSecret API," in ICAMIMIA 2015 - International Conference on Advanced Mechatronics, Intelligent Manufacture, and Industrial Automation, Proceeding - In conjunction with Industrial Mechatronics and Automation Exhibition, IMAE, 2016, pp. 181–183.

[175] C. Pham and T. Nguyen Thi Thanh, "Fresh food recognition using feature fusion," 2014 International Conference on Advanced Technologies for Communications (ATC 2014), Hanoi, 2014, pp. 298-302. doi: 10.1109/ATC.2014.7043401

[176] Dalakleidi, K., Sarantea, M. and Nikita, K. "A Modified All-and-One Classification Algorithm Combined with the Bag-of-Features Model to Address the Food Recognition Task" .In Proceedings of the 10th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2017) - Volume 5: HEALTHINF, pages 284-290, ISBN: 978-989-758-213-4

[177] G. Ciocca, P. Napoletano, and R. Schettini, “Food Recognition: A New Dataset, Experiments, and Results,” IEEE J. Biomed. Heal. Informatics, vol. 21, no. 3, pp. 588–598, 2017.

[178] J. Zheng, Z. Jane Wang, and C. Zhu, “Food image recognition via superpixel based low-level and mid-level distance Coding for smart home applications,” Sustain., vol. 9, no. 5, 2017.

[179] Y. He, N. Khanna, C. J. Boushey, and E. J. Delp, “Image segmentation for image-based dietary assessment: A comparative study,” in ISSCS 2013 - International Symposium on Signals, Circuits and Systems, 2013.

[180] P. F. Felzenszwalb and D. P. Huttenlocher, “Image segmentation using local variation,” Proceedings. 1998 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (Cat. No.98CB36231), 1998.

[181] D. Mery and F. Pedreschi, “Segmentation of colour food images using a robust algorithm,” J. Food Eng., vol. 66, no. 3, pp. 353–360, 2005.

[182] P. Pouladzadeh, S. Shirmohammadi, and A. Yassine, "Using graph cut segmentation for food calorie measurement," in IEEE MeMeA 2014 - IEEE International Symposium on Medical Measurements and Applications, Proceedings, 2014.

[183] F. Zhu, M. Bosch, N. Khanna, C. J. Boushey, and E. J. Delp, "Multiple hypotheses image segmentation and classification with application to dietary assessment," IEEE J. Biomed. Heal. Informatics, vol. 19, no. 1, pp. 377–388, 2015.

[184] J. Dehais, M. Anthimopoulos, and S. Mougiakakou, "Dish detection and segmentation for dietary assessment on smartphones," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2015, vol. 9281, pp. 433–440.

[185] J. Howe, "The Rise of Crowdsourcing," Wired Mag., vol. 14, no. 6, pp. 1–5, 2006.

[186] D. C. Brabham, "Crowdsourcing as a model for problem solving: An introduction and cases," Convergence, vol. 14, no. 1, pp. 75–90, 2008.

[187] A. Kittur, E. H. Chi, and B. Suh, "Crowdsourcing User Studies With Mechanical Turk," CHI '08 Proc. twenty-sixth Annu. SIGCHI Conf. Hum. factors Comput. Syst., pp. 453–456, 2008.

[188] M. K. Poetz and M. Schreier, "The value of crowdsourcing: Can users really compete with professionals in generating new product ideas?," J. Prod. Innov. Manag., vol. 29, no. 2, pp. 245–256, 2012.

[189] M. Vuković, "Crowdsourcing for enterprises," in SERVICES 2009 - 5th 2009 World Congress on Services, 2009, no. PART 1, pp. 686–692.

[190] G. Paolacci, J. Chandler, and P. Ipeirotis, "Running experiments on amazon mechanical turk," Judgm. Decis. Mak., vol. 5, no. 5, pp. 411–419, 2010.

[191] L. Johnson, C. Y. England, P. Laskowski, P. R. Woznowski, L. Birch, J. P. Hamilton-Shield, D. A. Lawlor, I. Craddock, and A. Skinner, “FoodFinder: developing a rapid low-cost crowdsourcing approach for obtaining data on meal size from meal photos,” Proceedings of the Nutrition Society, vol. 75, no. OCE3, p. E219, 2016.

[192] A. Moorhead, R. Bond and H. Zheng, "Smart food: Crowdsourcing of experts in nutrition and non-experts in identifying calories of meals using smartphone as a potential tool contributing to obesity prevention and management," 2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Washington, DC, 2015, pp. 1777-1779. doi: 10.1109/BIBM.2015.7359959

[193] G. M. Turner-McGrievy, E. E. Helander, K. Kaipainen, J. M. aria Perez-Macias, and I. Korhonen, “The use of crowdsourcing for dietary self-monitoring: crowdsourced ratings of food pictures are comparable to ratings by trained observers,” J. Am. Med. Inform. Assoc., vol. 22, no. e1, pp. e112–e119, 2015.

[194] G. M. Turner-McGrievy et al., “Crowdsourcing for self-monitoring: Using the Traffic Light Diet and crowdsourcing to provide dietary feedback,” Digit. Heal., vol. 2, no. 0, p. 205520761665721, 2016.

[195] L. I. Lesser, L. Wu, T. B. Matthiessen, and H. S. Luft, “Evaluating the healthiness of chain-restaurant menu items using crowdsourcing: A new method,” Public Health Nutr., vol. 20, no. 1, pp. 18–24, 2017.

[196] X. Chen and X. Yang, “Does food environment influence food choices? A geographical analysis through ‘tweets,’” Appl. Geogr., vol. 51, pp. 82–89, 2014.

[197] P. D. Howell, L. D. Martin, H. Salehian, C. Lee, K. M. Eastman, and J. Kim, “Analyzing Taste Preferences From Crowdsourced Food Entries,” in Proceedings of the 6th International Conference on Digital Health Conference - DH ’16, 2016, pp. 131–140.

[198] C. Szegedy, et al., Going deeper with convolutions, In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015, p. 19, vol. 7-12-2015.

[199] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, F.F. Li, Large-scale video classification with convolutional neural networks, In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2014, pp. 1725–1732.

[200] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553. pp. 436–444, 2015.

[201] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proc. IEEE, vol. 86, no. 11, pp. 2278–2323, 1998.

[202] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2014, pp. 512–519.

[203] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," Adv. Neural Inf. Process. Syst., pp. 1–9, 2012.

[204] C. Szegedy et al., "Going deeper with convolutions(Inception, GoogLeNet)," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015, vol. 07–12–June, pp. 1–9.

[205] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Int. Conf. Learn. Represent., pp. 1–14, 2015.

[206] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[207] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2014, pp. 580–587.

[208] R. Girshick, "Fast R-CNN," in Proceedings of the IEEE International Conference on Computer Vision, 2015, vol. 2015 International Conference on Computer Vision, ICCV 2015, pp. 1440–1448.

[209] A. Singla, L. Yuan, T. Ebrahimi, Food/Non-food image classification and food categorization using pre-trained GoogLeNet model, In: Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management - MADiMa 16, 2016, p. 311.

[210] H, K. Aizawa, M. Ogawa, Food Detection and Recognition Using Convolutional Neural Network, vol. 2, ACM Multimed, 2014, 10851088.

[211] M. Farooq, E. Sazonov, Feature extraction using deep learning for food type recognition, in: I. Rojas, F. Ortuo (Eds.), Bioinformatics and Biomedical Engineering: 5th International Work-conference, IWBBIO 2017, Granada, Spain, April 26–28, 2017, Proceedings, Part I, Springer International Publishing, Cham, 2017, pp. 464–472.

[212] Y. Kawano, K. Yanai, Food image recognition with deep convolutional features, ACM Int. Jt. Conf. Pervasive Ubiquitous Comput (2014) 589–593.

[213] L. Bossard, M. Guillaumin, and L. Van Gool, "Food-101 - Mining discriminative components with random forests," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2014, vol. 8694 LNCS, no. PART 6, pp. 446–461.

[214] F. Ragusa, V. Tomaselli, A. Furnari, S. Battiato and G.M. Farinella, Food vs non-food classification, In: Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management - MADiMa 16, 2016, 77–81.

[215] M. Farooq and E. Sazonov, Feature extraction using deep learning for food type recognition, In: I. Rojas and F. Ortuo, (Eds.), Bioinformatics and Biomedical Engineering: 5th International Work-conference, IWBBIO 2017, Granada, Spain, April 26–28, 2017, Proceedings, Part I, 2017, Springer International Publishing; Cham, 464–472.

[216] K. Yanai and Y. Kawano, Food image recognition using deep convolutional network with pre-training and fine-tuning, In: 2015 IEEE Interna- Tional Conference on Multimedia & Expo Workshops (ICMEW), 2015, 1–6.

[217] E. Aguilar, et al., Exploring Food Detection using CNNs, 2017, arXiv, 1709.04800v1 [cs], Sept.

[218] H. Hassannejad, G. Matrella, P. Ciampolini, I. De Munari, M. Mordonini, and S. Cagnoni, "Food Image Recognition Using Very Deep Convolutional Networks," in Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management - MADiMa '16, 2016, pp. 41–49.

[219] Ege, T., Yanai, K.: Estimating food calories for multiple-dish food photos. In: Proceedings of Asian Conference on Pattern Recognition (ACPR) (2017)

[220] E. Aguilar, M. Bolaños, and P. Radeva, "Exploring Food Detection Using CNNs," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2018, vol. 10672 LNCS, pp. 339–347.

[221] Kagaya and K. Aizawa, Highly Accurate Food/Non-food Image Classification Based on a Deep Convolutional Neural Network BT - New Trends in Image Analysis and Pro-

cessing – ICIAP 2015 Workshops: ICIAP 2015 International Workshops, September 7-8, 2015, BioFor, CTMR, RHEUMA, ISCA, MADiMa, SBMI, and QoEM; Genoa, Italy, Proceedings, V. Murino, E. Puppo, D. Sona, M. Cristani, and C. Sansone, Eds. Cham: Springer International Publishing, 2015, pp. 350–357.

[222] C. Cusano, P. Napoletano and R. Schettini, Evaluating color texture descriptors under large variations of controlled lighting conditions, J. Opt. Soc. Am. 33 (1), 2015, 17.

[223] J. A. Ello-Martin, J. H. Ledikwe, and B. J. Rolls, "The influence of food portion size and energy density on energy intake: implications for weight management.," The American journal of clinical nutrition, vol. 82, no. 1 Suppl. 2005.

[224] B. J. Rolls, E. L. Morris, and L. S. Roe, "Portion size of food affects energy intake in normal-weight and overweight men and women.," Am. J. Clin. Nutr., vol. 76, no. 6, pp. 1207–13, 2002.

[225] P. McAllister, H. Zheng, R. Bond, and A. Moorhead, "Semi-automated system for predicting calories in photographs of meals," in 2015 IEEE International Conference on Engineering, Technology and Innovation/ International Technology Management Conference, ICE/ITMC 2015, 2016.

[226] M. C. Carter, V. J. Burley, C. Nykjaer, and J. E. Cade, "My Meal Mate (MMM): Validation of the diet measures captured on a smartphone application to facilitate weight loss," Br. J. Nutr., vol. 109, no. 3, pp. 539–546, 2013.

[227] M. E. Rollo, S. Ash, P. Lyons-Wall, and A. W. Russell, "Evaluation of a mobile phone image-based dietary assessment method in adults with type 2 diabetes," Nutrients, vol. 7, no. 6, pp. 4897–4910, 2015.

[228] M. Vazquez-Briseno, C. Navarro-Cota, J. I. NietoHipolito, E. Jimenez-Garcia, and J. D. Sanchez-Lopez, "A proposal for using the internet of things concept to increase

children's health awareness," CONIELECOMP 2012, 22nd Int. Conf. Electr. Commun. Comput., pp. 168–172, Feb. 2012.

[229] E. Mattila, J. Pärkkä, M. Hermersdorf, J. Kaasinen, J. Vainio, K. Samposalo, J. Merilahti, J. Kolari, M. Kulju, R. Lappalainen, and I. Korhonen, "Mobile diary for wellness management–results on usage and usability in two user studies.," IEEE Trans. Inf. Technol. Biomed., vol. 12, no. 4, pp. 501–12, Jul. 2008

[230] S. Arivazhagan, R. N. Shebiah, S. S. Nidhyanandhan, and L. Ganesan, "Fruit Recognition using Color and Texture Features," Journal of Emerging Trends in Computing and Information Sciences 2010 vol. 1, no. 2, pp. 90–94

[231] G. Shroff, A. Smailagic, and D. P. Siewiorek, "Wearable context-aware food recognition for calorie monitoring," in Proceedings - International Symposium on Wearable Computers, ISWC, pp. 119– 120, 2008.

[232] M. Anthimopoulos et al., "Computer Vision-Based Carbohydrate Estimation for Type 1 Patients With Diabetes Using Smartphones," J. Diabetes Sci. Technol., vol. 9, no. 3, pp. 507–515, 2015.

[233] A. Myers et al., "Im2Calories: Towards an automated mobile vision food diary," in Proceedings of the IEEE International Conference on Computer Vision, 2015, vol. 2015 International Conference on Computer Vision, ICCV 2015, pp. 1233–1241.

[234] Yanchao Liang, Jianhua Li, "Computer vision-based food calorie estimation: dataset, method, and experiment", arXiv:1705.07632, 2017

[235] A. Direito, L. Pfaeffli Dale, E. Shields, R. Dobson, R. Whittaker, and R. Maddison, "Do physical activity and dietary smartphone applications incorporate evidence-based behaviour change techniques?," BMC Public Health, vol. 14, no. 1, 2014.

[236] G. Ciocca, P. Napoletano, and R. Schettini, “Food recognition and leftover estimation for daily diet monitoring,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2015, vol. 9281, pp. 334–341.

[237] Y. Yue et al., “Food volume estimation using a circular reference in image-based dietary studies,” in Proceedings of the 2010 IEEE 36th Annual Northeast Bioengineering Conference, NEBEC 2010, 2010.

[238] P. McAllister, H. Zheng, R. Bond, and A. Moorhead, “Towards personalised training of machine learning algorithms for food image classification using a smartphone camera,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016, vol. 10069 LNCS, pp. 178–190.

[239] P. McAllister, H. Zheng, R. Bond, and A. Moorhead, “A semi-automated food voting classification system: Combining user interaction and Support Vector Machines,” in International Symposium on Technology and Society, Proceedings, 2016, vol. 2016–March.

[240] P. McAllister, H. Zheng, R. Bond, and A. Moorhead, "Comparison of Machine Learning Algorithms in Classifying Segmented Photographs of Food for Food Logging", Proceeding of CERC 2016 Collaborative European Research Conference Cork Institute of Technology – Cork, Ireland 23 - 24 September 2016 www.cerc–conference.eu ISSN 2220 – 4164

[241] P. McAllister, H. Zheng, R. Bond, and A. Moorhead, “A digital technology framework to optimise the self-management of obesity,” in Proceedings of the 2016 ACM Interna- tional Joint Conference on Pervasive and Ubiquitous Computing Adjunct - UbiComp’16, 2016, pp. 1126–1131

[242] M. Bosch, F. Zhu, N. Khanna, C. J. Boushey, and E. J. Delp, “Food texture descriptors based on fractal and local gradient information,” in European Signal Processing Conference, 2011, pp. 764–768.

[243] "MATLAB - MathWorks", Mathworks.com, 2018. [Online]. Available: https://www.mathworks.com/prod [Accessed: 09- Mar- 2018].

[244] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, “The WEKA data mining software,” ACM SIGKDD Explor. Newsl., vol. 11, no. 1, p. 10, 2009.

[245] Java (Convolutional or Fully-connected) Neural Network Implementation, GitHub, 2017, [Online]. Available: https://github.com/amten/NeuralNetwork/releases/ tag/v1.1, [Accessed: 18- Sep- 2017].

[246] P. McAllister, H. Zheng, R. Bond, and A. Moorhead, “Combining deep residual network features with supervised machine learning algorithms to classify diverse food image datasets,” Comput. Biol. Med., Feb. 2018., DOI:10.1016/j.compbiomed.2018.02.008., ISSN: 00104825.

[247] P. McAllister, A. Moorhead, R. Bond and H. Zheng, "Automated adjustment of crowdsourced calorie estimations for accurate food image logging," 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Kansas City, MO, 2017, pp. 1059-1066. doi: 10.1109/BIBM.2017.8217803

[248] M. Tkalčič and J. F. Tasič, “Colour spaces - Perceptual, historical and applicational background,” in IEEE Region 8 EUROCON 2003: Computer as a Tool - Proceedings, 2003, vol. A, pp. 304–308.

[249] J. Ning, L. Zhang, D. Zhang, and C. Wu, “Interactive image segmentation by maximal similarity based region merging,” Pattern Recognit., vol. 43, no. 2, pp. 445–456, 2010.

[250] J. Cheng and J. C. Rajapakse, “Segmentation of clustered nuclei with shape markers and marking function,” IEEE Trans. Biomed. Eng., vol. 56, no. 3, pp. 741–748, 2009.

[251] H. Mo, B. Xu, W. Ouyang, and J. Wang, “Color segmentation of multi-colored fabrics using self-organizing-map based clustering algorithm,” Text. Res. J., vol. 87, no. 3, pp. 369–380, 2017.

[252] S. Dev, Y. H. Lee and S. Winkler, "Color-Based Segmentation of Sky/Cloud Images From Ground-Based Cameras," in IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 10, no. 1, pp. 231-242, Jan. 2017. doi: 10.1109/JSTARS.2016.2558474

[253] C. Zhu et al., “Retinal vessel segmentation in colour fundus images using Extreme Learning Machine,” Comput. Med. Imaging Graph., vol. 55, pp. 68–77, 2017.

[254] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-up robust features (SURF), Comput. Vis. Image Understand. 110 (3) (2008) 346359.

[255] N. Martinel, C. Piciarelli, C. Micheloni, G.L. Foresti, A structured committee for food recognition, In: Proceedings of the IEEE in- Ternational Conference on Com- puter Vision, 2015 February, 2015, pp. 484–492.

[256] H. Kagaya, K. Aizawa, and M. Ogawa, “Food Detection and Recognition Using Convolutional Neural Network,” in ACM Multimedia, 2014, no. 2, pp. 1085–1088.

[257] Y. Matsuda, H. Hoashi, K. Yanai, Recognition of multiple-food images by detecting candidate regions, In: Proceedings - IEEE International Conference on Multime- dia and Expo, 2012, p. 2530.

[258] Y. Kawano, K. Yanai, Automatic expansion of a food image dataset leveraging existing categories with domain adaptation, In: Lecture Notes in Computer Science

(Including Subseries Lecture Notes in Artificial Intel- Ligence and Lecture Notes in Bioinformatics), vol. 8927, 2015, p. 317.

[259] C. Cusano, P. Napoletano, R. Schettini, Local angular patterns for color texture classi cation, In: Lecture Notes in Computer Science (Includ- Ing Subseries Lecture Notes in Arti cial Intelligence and Lecture Notes in Bioinformatics), vol. 9281, 2015, pp. 111–118.

[260] G. M. Farinella, D. Allegra, and F. Stanco, "A Benchmark Dataset to Study the Representation of Food Images," in Computer Vision - ECCV 2014 Workshops, 2015, pp. 584–599.

[261] "Caltech101", Vision.caltech.edu, 2018. [Online]. Available: http://www.vision.caltech.edu/Image_Datas [Accessed: 11- Mar- 2018].

[262] A. Vedaldi, K. Lenc, MatConvNet, In: Proceedings of the 23rd ACM International Conference on Multimedia - MM 15, 2015, pp. 689–692.

[263] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2014, pp. 1701–1708.

[264] D. Ravi et al., "Deep Learning for Health Informatics," IEEE J. Biomed. Heal. Informatics, vol. 21, no. 1, pp. 4–21, 2017.

[265] Jia Deng, Wei Dong, R. Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," in 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.

[266] Scikit-learn Machine Learning in Python, Scikit-learn.org, 2017, [On- line]. Available: http://scikit-learn.org/, [Accessed: 24- Nov- 2017].

[267] G. James, D. Witten, T. Hastie, R. Tibishirani, An Introduction to Statistical Learning, 2013.

[268] J.P. Mueller, et al., Hitting Complexity with Neural Networks," Machine Learning for Dummies, Wiley, Hoboken, New Jersey, 2016279–290, ch. 16.

[269] GPU vs CPU in Convolutional Neural Networks Using TensorFlow — Relink, Relink, 2017, [Online]. Available: https://relinklabs.com/gpu- vs-cpu-in-convolutional-neural-networks-using-tensor ow, [Accessed: 19- Sep- 2017].

[270] Y. Kawano, K. Yanai, FoodCam-256: a Large-scale Real-time Mobile Food Recognition System Employing High-dimensional Features and Compression of Classi er Weights, 2014761–762, MM 14.

[271] D. C. Brabham, "Crowdsourcing as a Model for Problem Solving: An Introduction and Cases," Converg. Int. J. Res. into New Media Technol., vol. 14, no. 1, pp. 75–90, 2008.

[272] P. J. Stumbo, "New technology in dietary assessment: a review of digital methods in improving food record accuracy.," Proc. Nutr. Soc., vol. 72, no. 1, pp. 70–76, 2013.

[273] T. Joutou and K. Yanai, "A food image recognition system with Multiple Kernel Learning," Image Process. (ICIP), 2009 16th IEEE Int. Conf., pp. 285–288, 2009.

[274] Bosch, M., Zhu, F., Khanna, N., Boushey, C.J., Delp, E.J.: Combining global and local features for food identification in dietary assessment. In: Proceedings - Inter- national Conference on Image Processing, ICIP, pp. 1789–1792 (2011)

[275] Bosch, M., et. al.: Food texture descriptors based on fractal and local gradient information. In: 19th European Signal Processing Conference, pp. 764–768. IEEE (2011)

[276] Silva BVR, Cui J (2017), "A Survey on Automated Food Monitoring and Dietary Management Systems", J Health Med Informat 8:272. doi: 10.4172/2157-7420.1000272

[277] "File:Histograma bag of words.jpg - Wikimedia Commons", Commons.wikimedia.org, 2018. [Online].
Available: https://commons.wikimedia.org/wiki/File:Histograma_bag_of_words.jpg. [Accessed: 31- Jul- 2018].

[278] S. Gnanapriya, P. Esakkipriya, R. Kavipriya, C. Sangeethakamatchi and M. Sandhya, "Identification of organic fruits using color and size features," 2017 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR), Chennai, 2017, pp. 160-163. doi: 10.1109/TIAR.2017.8273707

[279] Da-Wen Sun, Cheng-Jin Du, Segmentation of complex food images by stick growing and merging algorithm, Journal of Food Engineering, Volume 61, Issue 1, 2004, Pages 17-26, ISSN 0260-8774, https://doi.org/10.1016/S0260-8774(03)00184-5.

[280] M. R. Marge, S. Banerjee, and A. I. Rudnicky, “Using the Amazon Mechanical Turk for Transcription of Spoken Language USING THE AMAZON MECHANICAL TURK,” in Acoustics, Speech and Signal Processing (ICASSP), 2010, pp. 5270–5273.

[281] R. Tanno, T. Ege, and K. Yanai, “AR DeepCalorieCam: An iOS App for Food Calorie Estimation with Augmented Reality,” in MultiMedia Modeling, 2018, pp. 352–356.

[282] Ege, T., Yanai, K.: Estimating Food Calories for Multiple-dish Food Photos. In: Proc. of Asian Conference on Pattern Recognition (ACPR). (2017)

[283] Ege, T., & Yanai, K. (2017). Simultaneous estimation of food categories and calories with multi-task CNN. In Proceedings of the 15th IAPR International Conference on Machine Vision Applications, MVA 2017. https://doi.org/10.23919/MVA.2017.7986835

[284] A. F. Subar et al., “Addressing Current Criticism Regarding the Value of Self-Report Dietary Data,” J. Nutr., vol. 145, no. 12, pp. 2639–2645, 2015.

[285] "Local binary patterns - File Exchange - MATLAB Central", Uk.mathworks.com, 2018. [Online]. Available: https://uk.mathworks.com/matlabcentral/fileexchange/36484-local-binary-patterns. [Accessed: 11- Oct- 2018].